STRESS ANALYSIS OF A SANDWICH STRCUTURE HAVING ISOTROPIC LAYERS USING FINITE DIFFERENCE METHOD

MD. KHAIRUL HABIB PULOK

DEPARTMENT OF MECHANICAL ENGINEERING

BANGLADESH UNIVERSITY OF ENGINEERING & TECHNOLOGY

DHAKA-1000, BANGLADESH.

JULY, 2016

STRESS ANALYSIS OF A SANDWICH STRCUTURE HAVING ISOTROPIC LAYERS USING FINITE DIFFERENCE METHOD

by

Md. Khairul Habib Pulok

A thesis submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

Department of Mechanical Engineering
BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY
Dhaka 1000, Bangladesh.

July, 2016

This thesis titled "**STRESS ANALYSIS OF A SANDWICH STRCUTURE HAVING ISOTROPIC LAYERS USING FINITE DIFFERENCE METHOD",** submitted by **Md. Khairul Habib Pulok**, Student No. **0412102023P,** Session: **April 2012**, has been accepted as satisfactory in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE IN MECHANICAL ENGINEERING on July 24 , 2016.

BOARD OF EXAMINERS

Chairman (Supervisor)

____
Dr. Md. Abdus Salam Akanda
Professor
Department of Mechanical Engineering
BUET, Dhaka, Bangladesh.

Member (Ex-Officio)

____
Dr. Mohammad Ali
Professor and Head
Department of Mechanical Engineering
BUET, Dhaka, Bangladesh.

Member

____
Dr. Shaikh Reaz Ahmed
Professor
Department of Mechanical Engineering
BUET, Dhaka, Bangladesh.

Member (External)

____
Dr. Khan Mahamud Amanat
Professor
Department of Civil Engineering
BUET, Dhaka, Bangladesh.

CERTIFICATE OF RESEARCH

This is to certify that the work presented in this thesis is carried out by the author under the supervision of Dr. Md. Abdus Salam Akanda, Professor of the Department of Mechanical Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh.

_____________________________ _________________________

Dr. Md. Abdus Salam Akanda Md. Khairul Habib Pulok

DECLARATION

It is hereby declared that this thesis or any part of it has not been submitted elsewhere for the award of any degree or qualification.

Md. Khairul Habib Pulok

Author

Date

Dedicated to My Parents

ACKNOWLEDGEMENTS

Thanks to most merciful, most gracious and the most cordial Almighty Allah. The author would like to thank his mother for giving him unquestionable support and inspirations.

The author would like to express his gratefulness and deepest reverence to his thesis supervisor Dr. Md. Abdus Salam Akanda, Professor, Department of Mechanical Engineering, BUET for his constant guidance, cynosure, constructive criticism, encouragement and careful supervision throughout the research work without which this thesis would not have egress.

The author gratefully acknowledges the assistance received at various stages of this work from Dr. Shaikh Reaz Ahmed, Professor, Department of Mechanical Engineering, BUET.

Special thanks to all his teachers, especially Dr. Mohammad Abdul Motalab and Mr. Aminul Islam Khan for giving support and encouragement. The author expresses thankfulness to his siblings for their encouragement and inspirations.

Finally, the author would like to express his gratitude to the other students performed their thesis under same supervisor especially M. Shafiqur Rahman for his cooperation and support in the successful completion of this work.

Author

ABSTRACT

This thesis deals with the stress analysis of sandwich structured composite materials. Materials under consideration are assumed to be perfectly bonded together. Finite difference method is used for the solution of two dimensional elastic problems. A numerical model for rectangular geometry based on displacement potential function has been developed to investigate the problem. In each layer of the composite the mechanical properties are assumed to be isotropic.

At the interface, there is a single value for each displacement component but two different values for each stress component of the laminated composite having different mechanical properties in layers. Like usual critical zone of a sandwich structured composite under mechanical loading, the two interfacial zones are also zone of critical stresses. Changing the Poisson's ratio in any layer (case or core) has significant effects on the results of all layers of the sandwich structured composite. Due to the mathematical expressions of stresses and displacements for two dimensional elastic problems, the study of the effects of Poisson's ratio is intricate rather the study of the effects of Modulus of elasticity is straightforward. In general, the material having higher modulus of elasticity experiences higher stresses.

Finite difference scheme has been developed for the management of boundary conditions so that all possible mixed boundary conditions can be applied in any boundary points as well as at the interfaces of isotropic layers. Special numerical formulations yield to new formula structures are employed at the interfaces. An effective programming code has been developed by FORTRAN language to solve the problems of sandwich structured composites. In order to compare the results by the present finite difference method, another numerical technique namely finite element method is used. Validation of the results is performed by using commercially available FEM package software. It is observed that the results agree well within the acceptable limit, which also confirms to the reliability of the finite difference method.

TABLE OF CONTENTS

NOMENCLATURE

Notation	**Definition**
E	Modulus of Elasticity
μ	Poisson ratio
ψ	Displacement potential function
σ_x	Normal stress component along x-direction
σ_y	Normal stress component along y-direction
σ_z	Normal stress component along z-direction
σ_{xy} or τ_{xy}	Shear stress component in the xy plane
σ_n	Stress component normal to boundary
σ_t	Stress component tangential to boundary
ε_x ,ε_y ,ε_z	Strain components parallel to the co-ordinate axes.
u_x or u	Displacement component along x-direction
u_y or v	Displacement component along y-direction
$\varphi(x,y)$	Airy's stress function
h	Mesh length along x-direction
k	Mesh length along y-direction
r	k/h
a	Length of the body
b	Width of the body
l, m	Direction cosine of the normal at any physical boundary point

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

1.1 General

At the present time composite is a very common word because of its multipurpose application in many industries such as aerospace, automotive, marine, construction etc. The word “composite” means ‘consisting of two or more distinct parts’. Composites are formed by combining materials together to form an overall structure that is better than the individual components. The constituent materials have significantly different physical or chemical properties, that when combined, produce a material with characteristics different from the individual components. The individual components remain separate and distinct within the finished structure.

The use of composite materials is gradually increasing day-by-day, especially, to satisfy the demand of lightweight structures. The use of short composite columns or struts in the construction of engineering structures and machines is quite extensive. It is known that the mechanical properties like, strength, toughness of a fiber reinforced composite differ significantly from those of the isotropic materials, which eventually play an important role in defining the state of stress and displacement of the corresponding structure under loading [1].

A sandwich-structured composite is a special class of composite materials that is usually fabricated by attaching two thin but stiff layers to a lightweight but thick core. The core material is normally low strength material, but its higher thickness provides the sandwich composite with high bending stiffness with overall low density. Sandwich panels are used in applications where high structural rigidity and low weight are required. Sandwich-structured composite are used in aircrafts, where mechanical performance and weight saving is essential.

Almost all structures consist of assembly of simple elements which are connected to each other by joints. Joints or connections that are made in the composite structures can be broadly divided into two categories: adhesively bonded and mechanically fastened using bolts or rivets. Adhesive bonding has been applied successfully in many technologies. [2-3].

Cladding, the application of one material over another to provide a skin or layer is another process to form a sandwich-structured composite. In cladding, a metal coating bonded onto another metal under high pressure and temperature.

Anodic bonding is a bonding procedure without any intermediate layer. This bonding technique, also known as field assisted bonding or electrostatic sealing, is mostly used for connecting silicon/glass and metal/glass through electric fields [4]. In this bonding two layers are connected at the molecular level. In this type of bonding there is no slip at the interface [5].

1.2 Background of the study

Because of the necessity of utilizing (existing and prospective) sandwich-structured materials in many engineering fields, it is of great importance to understand the behavior of the sandwich-structured composite. For the solution of the problem several methodologies could be followed, however, all of the methods could be classified into three general categories: Experimental, analytical and numerical method. Though experimental methods give the most reliable results, it requires special equipments, testing facilities, thus, very costly. Analytical solution of every problem is almost impossible because of complex boundary conditions and geometries. For this reason the numerical methods had become the ultimate choice by the researchers in the last few decades. Invention and rapid improvement of the computing machine, i.e. sophisticated high performance computers have accelerated the popularity of the numerical methods.

Stress analysis of sandwich-structured composite requires the solution of partial differential equations. There are various numerical methods available for the solution of partial differential equations. Among them the most popular methods are Finite Element Method (FEM) and Finite Difference Method (FDM). Finite difference method is an ideal numerical approach for solving partial differential equations. The difference equations that are used to model governing equations in FDM are very simple to code and the global coefficient matrix produced by FDM is a banded structure and is very effective to good solution. In spite of these characteristics, now-a-days, finite difference method is replaced in most of the engineering applications by the finite element method. Because finite element method is very efficient at managing complex boundary shapes and produces reliable result within the body of the structure. It is noted that critical stresses occur most frequently at the boundary of the structures and Dow et al. [6] verified that the accuracy of finite difference method in reproducing the state of stresses along the boundary surfaces was much higher than that of the finite element method. Not only that, Hossain et al. [7] showed that if an efficient approach is developed based on finite difference method, the computational effort is greatly reduced as compared to other methods.

The present work is confined to isotropic, homogeneous, and elastic properties in each layer of the sandwich-structured composite. This work considers every (three) layers of sandwich structured composite as isotropic and homogeneous solid material.

Almost all engineering materials possess to a certain extent the property of elasticity. The response of a solid body to the external load is influenced by the geometric configuration of the body as well as the mechanical properties of the material. In this study the mechanical properties are different in casing and core of sandwich-structured composite.

1.3 Objectives of the present study

The specific objectives of the present research work are as follows:

(a) Development of a finite difference scheme with special treatment of governing equation and boundary conditions for interfaces of sandwich-structured composite.

(b) Numerical study of 2D elastic problem of sandwich-structured composite by finite difference method.

(c) To determine the displacement and stress distribution in the layers as well as at the interfaces of the elastic problem.

(d) To compare the results obtained by the finite difference and finite element methods.

1.4 Literature Review

First application of finite difference equations, i.e. numerical method, in elasticity was done by Runge [8], who used this method in solving torsional problems. Subsequently finite difference method found very wide application in publications of stress analysis. Successful application of the stress function in conjunction with the finite difference method was reported in 1951 by Conway et al. [9]. The main shortcoming of the stress function formulation is that it accepts boundary conditions in terms of boundary loadings only. So problems containing boundary conditions in terms of restraints only or in terms of both loading and restraints (mixed boundary value problems) could not be solved by this stress function formulation. With a view to solving the problems of mixed boundary conditions, Uddin [10] proposed a formulation for the solution of two dimensional such mixed boundary value problems using the displacement potential function formulation and successfully applied this formulation for the solution of many two dimensional elastic mixed boundary value problems [11-16]. Not only that, Hossain [17] extended the displacement potential function formulation for three dimensional elastic problems and obtained reliable solution for some classical problems of solid mechanics [18]. Solution of the two dimensional elastic problem with hole is successfully carried out by Rahman [19]. However, these works are limited to single isotropic material only. Beside the finite difference method, another numerical method namely finite element method was first successfully applied for the two dimensional elastic problem by Turner et al. [20] and Clough [21]. Afterwards it became very popular and reliable with the rapid development of the digital computers and used by researchers in both two dimension and three dimension [22].

Long et al. [23] predicted the nominal stress-strain curves of a multi-layered composite material by FE Analysis. Sevecek et al. [24] analytically performed stress-strain analysis of the laminates with orthotropic (isotropic) layers using Classical Laminate Theory and compared it with finite element analysis considering the thermal loading. Some other researchers have used finite element technique for stress analysis of some layered materials [25-27]. Arbaoui et al. [28] Analyzed numerical simulation and experimental bending behavior of multi-layered sandwich-structures. Problems with various mechanical loadings were not present in these studies.

Later, the displacement potential function approach of the finite difference method had been extended for investigating bond-line stresses of tire tread section by Sankar et al. [29] and determination of the stresses for composite lamina considering directional mechanical properties was performed by Alam et al. [30]. After that, Bhuiyan [31] extended the finite difference technique with displacement potential function approach for a different type of composite made of two bonded isotropic materials. Therefore, stress analysis of a sandwich-structured material is yet to be attempted by this approach.

From the above survey it is evident that, the present study of finding state of stress and displacement in sandwich-structured composite for various mechanical loadings is not only an interesting practical subject, but also of great importance because of its presence and prospect in many structural components. Application of finite difference technique based on displacement potential function for the solution of stresses of sandwich-structured composite will be a new attempt to extend the capability of displacement potential formulation.

1.5 Scope of the present study

This thesis consists of five chapters. A detail literature review is provided in Chapter 1. It illustrates the earlier research works showing various achievements and events that occurred in the field of stress analysis, theory of elasticity, application of various solution techniques and the evolution of sandwich-structured composite material research. In chapter 2, the relevant basic theories are stated in brief to understand the theories of elasticity. The mathematical model, used for the finite difference scheme of the study is described for better understanding of displacement potential function formulation.

Chapter 3 depicts the numerical modeling of the problem, mainly the finite difference formulation of the fourth order partial differential governing equation and different boundary conditions. This chapter also describes the treatment of the formulation of boundary conditions. Later in the chapter, a summary is also made on finite element method since finite element method is the supporting tool for validation of finite difference results of the study.

In chapter 4, detailed analysis of results is presented accompanied by a validation. A similar problem is solved by the FDM and FEM. Both results are compared with each other. Results obtained from the finite difference technique for different boundary conditions are critically analyzed. Effects of different combination of poisson's ratios and modulus of elasticity are analyzed.

Finally in Chapter 5, the exposition is completed with the main conclusions and the recommendations for future works.

CHAPTER 2
MATHEMATICAL MODEL

2.1 Introduction

Almost all engineering materials possess the property of elasticity to a certain extent. The external forces producing deformations do not exceed certain limit which is called the elastic limit and the deformation disappears as the removal of forces within this limit. In the analysis, it will be assumed that the sandwich-structured composite materials undergoing the action of mechanical loadings are perfectly elastic and the deformations are very small. So to apply and understand the theories of elasticity completely, it is necessary to introduce some basic terms related to the theory of elasticity.

2.2 Stresses at a Point

Under the action of external forces, internal forces are produced within the elastic body. The intensity i.e. internal forces per unit area of the surface on which they act is called stress. External forces may be of two types: surface force and body force. Forces distributed over the surface of a body, such as hydrostatic pressure, are called surface forces. Forces distributed over the volume of the body, such as gravitational force or inertia force, are called body force. As the effect of body forces as compared to the surface forces is very small, in most practical cases body forces are neglected. In the present study only the surface forces are taken into consideration.

The displacements, strains and stresses in a deformable body are interlinked. Additionally, they all depend on the geometry and material of the work piece, external forces and supports. The discussion is beginning on the governing equations with the concept of stress at a point. To understand the concept of stress at a point, consider a body subjected to external forces and supported in a suitable fashion, as shown in Figure 2.1. Note that, as soon as the forces are applied, the body gets deformed and sometimes displaced if the supports do not restrain the rigid body motion of the body. Thus, Figure 2.1 shows the deformed configuration. In fact, throughout this section, the configuration considered will be the deformed configuration. First, the stress vector (on a plane) is defined at point P of the body. For this, a plane (called as cutting plane) is passed through point P having a unit normal **n**. On each half of the body, there are distributed internal forces acting on the cutting plane and exerted by the other half. On the left half, a small area ΔA is considered around point P of the cutting plane. Let **ΔF** be the resultant of the distributed internal forces (acting on ΔA) exerted by the right half. Then, the stress vector (or traction) at point P (on the plane with normal **n**) is defined as

$$\boldsymbol{t}_n = \lim_{\Delta A \to 0} \frac{\Delta \boldsymbol{F}}{\Delta A} \tag{2.1}$$

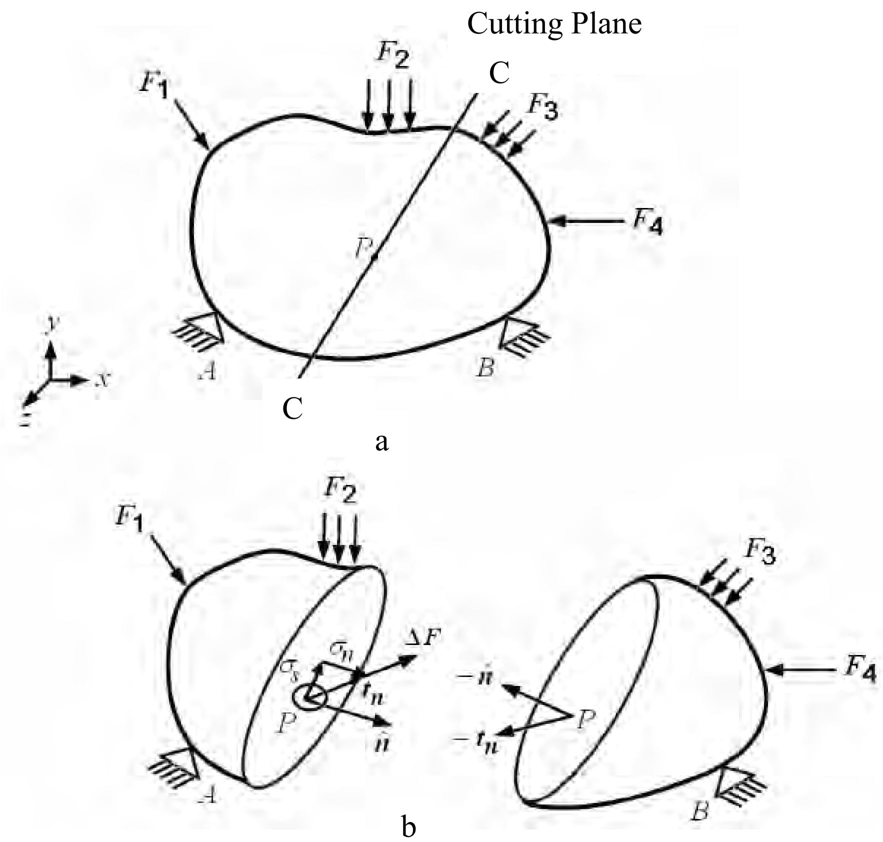


Figure 2.1: Stress vector at a point on a plane CC. (a) Cutting plane passing through point P of the deformed configuration, (b) Stress vector $\mathbf{t_n}$, normal stress component σ_n and shear stress component σ_s acting at point P on the cutting plane

The component of $\mathbf{t_n}$ normal to the plane is called as the normal stress component. It is denoted by σ_n and The component of $\mathbf{t_n}$ along the plane is called as the shear stress component. It is denoted by σ_s. Note that, on the right half, the normal to the cutting plane will be **-n** and the stress vector at P will be $-\mathbf{t_n}$ as per the Newton's third law.

It can be shown that a stress vector on any arbitrary plane can be uniquely represented in terms of the stress vectors on three mutually orthogonal planes. To show this, we consider x, y and z planes as the three planes, having normal vectors along the three Cartesian directions x, y and z respectively. Let the stress vectors on x, y and z planes be denoted by $\mathbf{t_x}$, $\mathbf{t_y}$ and $\mathbf{t_z}$ respectively. Further, we denote their components along x, y and z directions as follows [48]:

$$\mathbf{t_x} = \sigma_{xx}\,.\mathbf{i} + \sigma_{xy}.\mathbf{j} + \sigma_{xz}.\mathbf{k} \qquad (2.2)$$

$$\mathbf{t_y} = \sigma_{yx}\,.\mathbf{i} + \sigma_{yy}.\mathbf{j} + \sigma y_{z}.\mathbf{k} \qquad (2.3)$$

$$\mathbf{t_x} = \sigma_{zx}\,.\mathbf{i} + \sigma_{zy}.\mathbf{j} + \sigma_{zz}.\mathbf{k} \qquad (2.4)$$

where, (**i, j ,k**) are the unit vectors along(x, y ,z) axes. The stress vectors and their components are shown in Figure 2.2.To derive the above result, we consider a small element at point P whose shape is that of a tetrahedron. The three sides of the tetrahedron are chosen perpendicular to x, y and z axes and the slant face is chosen normal to vector **n**. Then, equilibrium of the tetrahedron in the limit as its size goes to zero leads to the following result:

$$\mathbf{t_n} = \mathbf{t_x}\,.n_x + \mathbf{t_y}.\,n_y + \mathbf{t_z}.n_z \qquad (2.5)$$

where, n_x, n_y, and n_z are the components of the normal vector **n**. This result is true for every stress vector at point P no matter what the orientation of the normal vector **n** is. Further, this result remains valid even if the body forces are not zero or the body is accelerating.

Let the components of the stress vector **t_n** be

$$\mathbf{t_n} = (t_n)_x.\ \mathbf{i} + (t_n)_y.\ \mathbf{j} + (t_n)_z.\ \mathbf{k} \tag{2.6}$$

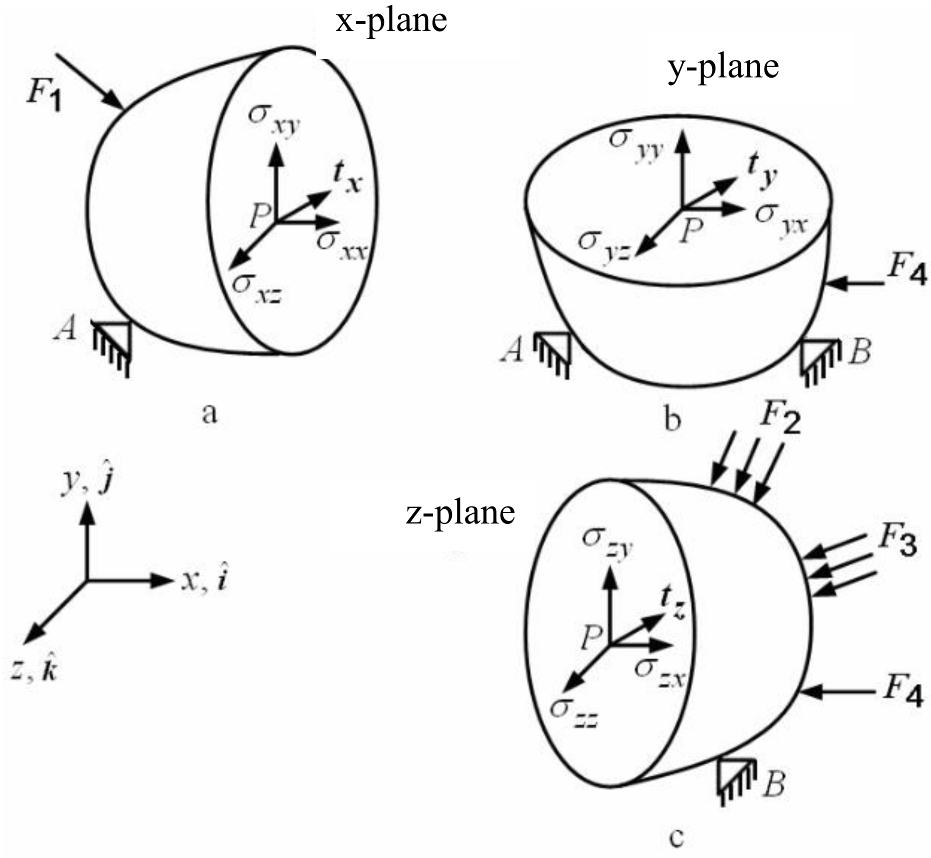


Figure 2.2: Stress vectors and their components on x, y, and z plane. (a) Stress vector and its components on x plane, (b) Stress vector and its components on y plane, (c) Stress vector and its components on z plane

Substituting Eqs. (2.2-2.4) and (2.6), we get the component form of Eq. (2.5) as follows:

$$\begin{Bmatrix} (t_n)_x \\ (t_n)_y \\ (t_n)_z \end{Bmatrix} = \begin{bmatrix} \sigma_{xx} & \sigma_{yx} & \sigma_{zx} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{zy} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{bmatrix} . \begin{Bmatrix} n_x \\ n_y \\ n_z \end{Bmatrix} \tag{2.7}$$

In array notation, this can be written as

$$\{t_n\} = [\sigma]^T . \{n\} \tag{2.8}$$

Where, the stress matrix $[\sigma]$ is

$$[\sigma] = \begin{bmatrix} \sigma_{xx} & \sigma_{yx} & \sigma_{zx} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{zy} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{bmatrix} \tag{2.9}$$

Therefore, it is evident that the stress at a point can be completely described by means of just three stress vectors $\mathbf{t_x}$, $\mathbf{t_y}$ and $\mathbf{t_z}$ acting on mutually orthogonal planes or by their nine components: σ_{xx}, σ_{yy}, σ_{zz}, σ_{xy}, σ_{yx}. σ_{yz}, σ_{zy}, σ_{xz}, and σ_{zx}. In the notation of stresses, the first index describes the direction of the normal to the plane on which the stress component acts while the second index represents the direction of the stress component itself. Thus, σ_{xy} indicates a stress component acting in y -direction on x -plane.

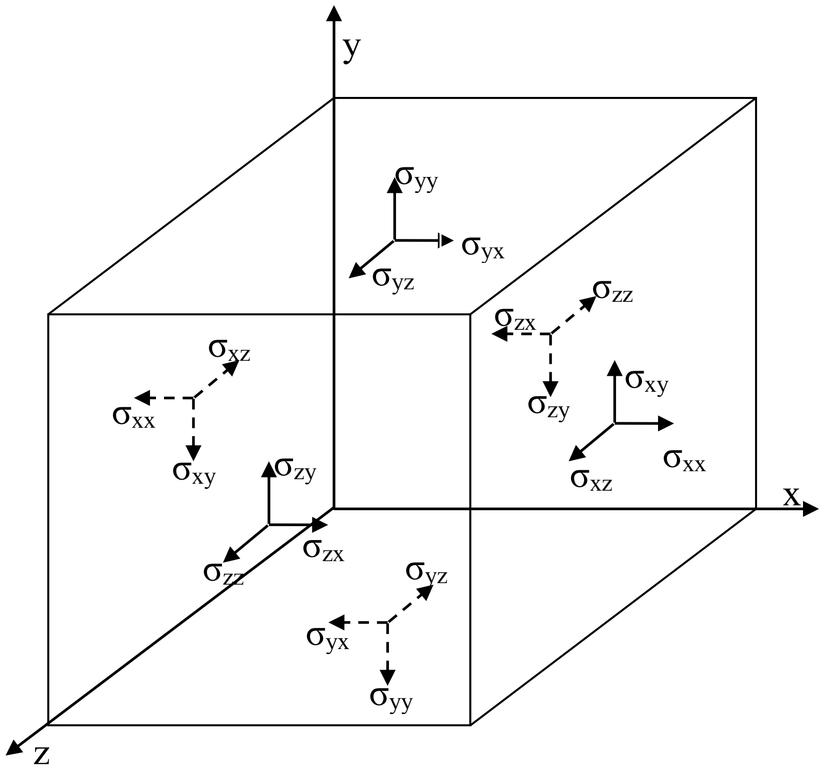


Figure 2.3: Stress components in a cubic element

When both the indices are same, it means the stress component is along the normal to the plane on which it acts. It is called as the normal stress component. Thus, σ_{xx} , σ_{yy} and σ_{zz} are the normal stress components. When the two indices are different, it means the direction of the component is within the plane. Such a component is called as the shear stress component. These components could be better understood with reference to a cubic element as shown in Figure 2.3. Based on the consideration of the static equilibrium of the element it could be shown that $\sigma_{xy}= \sigma_{yx}$, $\sigma_{yz}= \sigma_{zy}$, and $\sigma_{xz}= \sigma_{zx}$. As a result, the nine components of stress are reduced to six independent components only.

2.3 Sign Convention for Stresses

In the present study, the following sign convention is adopted for the stress components. First positive and negative planes have been defined. A plane i is considered positive if the outward normal to it points in the positive i direction, otherwise it is considered as negative. A stress component is considered positive if it acts in positive direction on positive plane or in negative direction on negative plane. Otherwise, it is considered as negative. Figure 2.4 illustrates positive and negative normal and shear stress components. Throughout the finite difference analysis of the present problem, this sign convention has been used.

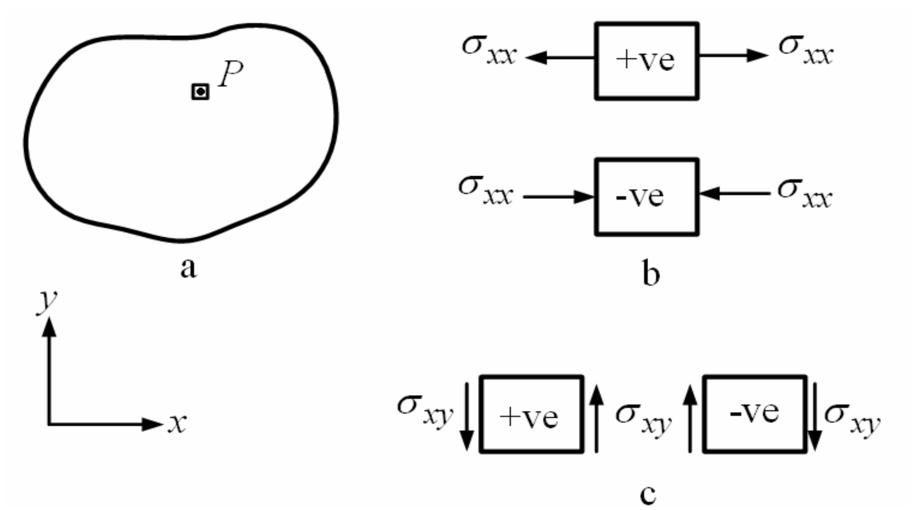


Figure 2.4: Sign convention for normal and shear stress components (a) Small element at point 'P' in the deformed configuration. Forces on the body and supports are not shown, (b) Positive and negative 'σ_{xx}', (c) Positive and negative 'σ_{xy}'

2.4 Stress-Strain Relationship

Deformation at a point is related to the displacement of the neighborhood of that point. The neighborhood of a point is defined as a set of points in the close vicinity of that point. The displacement consists of three parts: (i) displacement due to translation of the neighborhood of that point, (ii) displacement due to rotation of the neighborhood of that point and (iii) displacement due to deformation of the neighborhood of that point. In this analysis the deformations of the elastic body is considered very small hence, the deformation is elastic. The state of strain at any point could be completely defined by six components of strain: ε_x, ε_y, ε_z, γ_{xy}, γ_{yz}, and γ_{zx}.

By definition the normal and shear strain can be given by [31]

$$\varepsilon_x = \frac{\partial u_x}{\partial x}, \quad \varepsilon_y = \frac{\partial u_y}{\partial y}, \quad \varepsilon_z = \frac{\partial u_z}{\partial z} \tag{2.10}$$

$$\gamma_{xy} = \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x}, \ \gamma_{yz} = \frac{\partial u_y}{\partial z} + \frac{\partial u_z}{\partial y}, \ \gamma_{zx} = \frac{\partial u_x}{\partial z} + \frac{\partial u_x}{\partial z} \tag{2.11}$$

where, ε_x, ε_y, ε_z are the strain components parallel to the coordinate axes called normal strain and γ_{xy}, γ_{yz}, γ_{zx} are strain components acting on the planes xy, yz and zx planes respectively called shear strain.

The stresses are related to the strains by the Hooke's law. The generalized Hooke's law suggests that each of the stress components is the linear function of the strain components. The stresses are related to the strains by the Hooke's law and Poisson's law as follows [14]:

$$\varepsilon_x = \frac{1}{E}\left[\sigma_x - \mu(\sigma_y + \sigma_z)\right]$$

$$\varepsilon_y = \frac{1}{E}\left[\sigma_y - \mu(\sigma_x + \sigma_z)\right] \tag{2.12}$$

$$\varepsilon_z = \frac{1}{E}\left[\sigma_z - \mu(\sigma_x + \sigma_y)\right]$$

where, E is the modulus of elasticity and μ is the poisson's ratio.

2.5 Plane Stress and Plane Strain

Although practically all the bodies are three dimensional, most of the practical problems of stress analysis could be reduced to two dimensional under two simplifying assumptions. One, the loading on the body is confined in a plane and the dimension of the body in the direction perpendicular to this plane is relatively small as compared to the others. In such cases, the stresses in the body perpendicular to the plane of loading are usually very small and thus can be neglected. As a result these problems become two dimensional, usually referred to as plane stress problems. Two, one of the three dimensions of the body is relatively large or straining in a particular direction is restrained. In such cases, the stresses in the large or restrained direction are zero. As a result these problems become two dimensional and usually referred to as plane strain problems.

If a thin plate is loaded by forces applied at the boundary, parallel to the plane of the plate and distributed over the thickness (Figure 2.5), the stress components σ_{zz}, σ_{zx}, σ_{yz} become zero on both faces of the plate and it may assumed that they are also zero within the plate. Thus in a plane stress problems the state of stress is defined by σ_{xx}, σ_{yy}, σ_{xy} only.

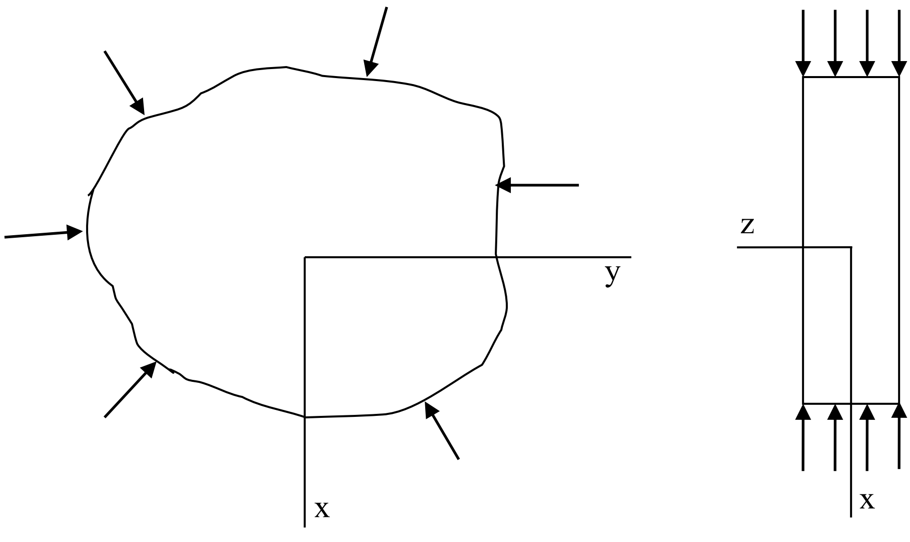


Figure 2.5: Plane stress problem

A similar simplification is possible when the dimension of the body in the z-direction is very large. If a long cylindrical or prismatic body is loaded by forces that are perpendicular to the longitudinal elements and do not vary along the length, it may be assumed that all the cross sections are in the same condition.

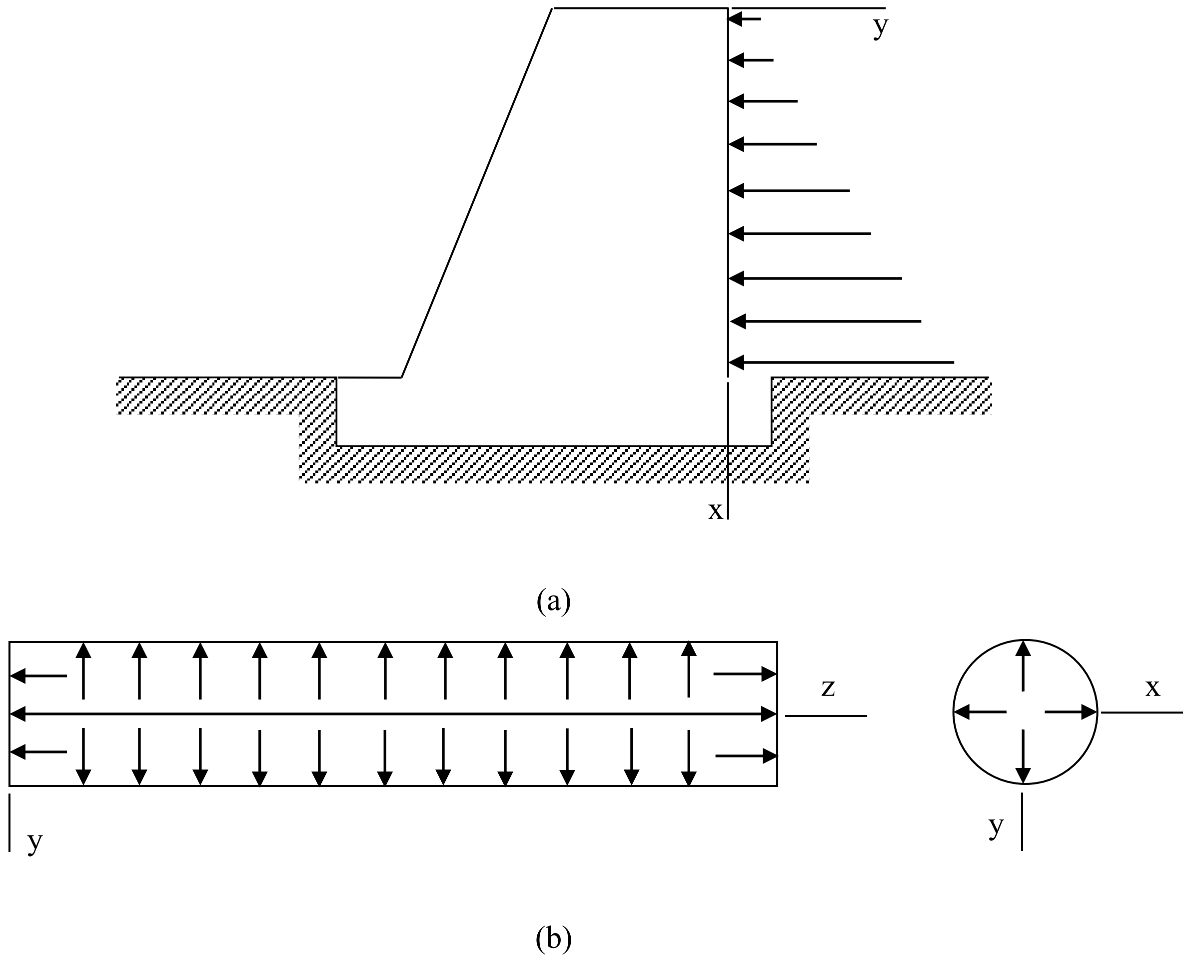


Figure 2.6: Plane Strain problem

Problems like a retaining wall with lateral pressure (Figure 2.6a), a culvert, cylindrical tube with internal pressure (Figure 2.6b) etc. can be considered as the plane strain problems. In plane strain problems ε_z, γ_{yz}, γ_{zx} are zero and thus the stress components σ_{zx}, σ_{yz} become zero. Thus the state of stress is defined by σ_{xx}, σ_{yy}, σ_{xy} only.

2.6 Differential Equations of Equilibrium and Boundary Conditions

For static equilibrium of the infinitesimal cubic element as shown in Figure 2.3 the following equations can be obtained, [31]

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z} + X = 0$$

$$\frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial z} + Y = 0 \qquad (2.13)$$

$$\frac{\partial \sigma_{zz}}{\partial z} + \frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} + Z = 0$$

These equations (2.13) are known as the equations of equilibrium, where X, Y, and Z are the components of body force per unit volume of the element in x, y, and z-directions respectively. The body forces can be eliminated due to their negligible effect as compared to that of surface forces. For plane stress condition the cubic element reduces to a thin rectangular block and no body forces acting on that block, hence the equilibrium equations yields to

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} = 0$$

$$\frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{xy}}{\partial x} = 0 \qquad (2.14)$$

Above equations must be satisfied at all points throughout the body. The stress components vary over the volume of the block. At the boundary they must be in equilibrium with external forces on the boundary and the external forces may be considered as the continuation of the internal stress distribution. So the conditions of equilibrium at the boundary can be written as [31],

$$\sigma_n = \sigma_{xx}.\, l^2 + \sigma_{yy}.\, m^2 + 2\sigma_{xy}.\, l\, m$$

$$\sigma_t = \sigma_{xy}.\, (l^2 - m^2) + (\sigma_{yy} - \sigma_{xx}).\, l\, m \qquad (2.15)$$

where, σ_n and σ_t are the normal and tangential components of the surface forces acting on the boundary per unit area and l, m are the direction cosines of the normal to the surface.

Similarly, normal component of displacement u_n and the tangential component u_t acting on the boundary surface can be expressed by

$$u_n = u_x.l + u_y.m$$

$$u_t = u_y.l - u_x.m \qquad (2.16)$$

Generally normal components (σ_n and u_n) are considered to be positive when act outward on the boundary and the tangential components (σ_t and u_t) are considered positive if they act in the anti-clockwise direction on the body.

2.7 Compatibility Equations

To determine the state of stress in the two-dimensional elastic body, it is necessary to find the solution of the equilibrium equations (Eq. 2.14), which must satisfy the boundary conditions (Eq. 2.15 and 2.16) at the boundary. Since these two equations contain three unknown stress components (σ_{xx}, σ_{yy}, and σ_{xy}), they are not sufficient to determine the three components. Therefore, the problem is a statically indeterminate one. As a result, to obtain the solution, the elastic deformations of the body must be taken into consideration. For two dimensional body three strain components can be expressed in terms of the displacement components as

$$\varepsilon_x = \frac{\partial u_x}{\partial x}; \qquad \varepsilon_y = \frac{\partial u_y}{\partial y}; \qquad \gamma_{xy} = \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \qquad (2.17)$$

Since these three strain components are expressed by two functions only, they cannot be related arbitrarily among themselves. There exists a certain relationship among the strain components, which is expressed as,

$$\frac{\partial^2 \varepsilon_x}{\partial y^2} + \frac{\partial^2 \varepsilon_y}{\partial x^2} = \frac{\partial^2 \gamma_{xy}}{\partial x.\partial y} \qquad (2.18)$$

This differential relation is called the condition of compatibility. It must be satisfied by the strain components to ensure the existence of functions σ_x and u_y connected with the strain components by Eq. 2.17

Elimination of strains in terms of stresses, equation 2.18 yields to

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)(\sigma_x + \sigma_y) = 0 \qquad (2.19)$$

The equations (Eq. 2.14) of equilibrium together with the boundary conditions (Eq. 2.15) and the above compatibility equation (Eq. 2.19) give us a system of equations that is usually sufficient for the complete solution of stress distribution in a two dimensional problem.

2.8 Solution Technique for 2-D Problems with Known Stresses at the Boundary

The solution of two dimensional elastic problems requires integration of the differential equations of equilibrium (Eq. 2.14) together with the compatibility equations (Eq. 2.19) and the boundary conditions (Eq. 2.15).

The usual method of solving these equations is through the introduction of a function φ(x,y), known as Airy stress function, defined as

$$\sigma_x = \frac{\partial^2 \phi}{\partial y^2}, \qquad \sigma_y = \frac{\partial^2 \phi}{\partial x^2}, \qquad \sigma_{xy} = \frac{\partial^2 \phi}{\partial x . \partial y} \qquad (2.20)$$

which satisfies equations (Eq. 2.14) and transforms the equation (Eq. 2.19) into

$$\frac{\partial^4 \phi}{\partial x^4} + 2\frac{\partial^4 \phi}{\partial x^2 . \partial y^2} + \frac{\partial^4 \phi}{\partial y^4} = 0 \qquad (2.21)$$

Ultimately, equation (Eq. 2.21) has to be integrated satisfying equation (Eq. 2.15) at the boundary. But the solution approach stated above through the stress function φ(x,y) is a special case of a general problem. Only a problem with pure known stress at the boundary can be solved by this approach. But, most of the practical engineering problems are with the mixed boundary conditions, that is, the conditions at the boundary might include known stresses, known displacements or combination of stresses and displacements with different conditions in different segments at the boundary. A problem of this kind can not be solved through Airy's stress function φ(x,y), defined in equation (Eq. 2.20).

2.9 Mathematical Formulation in terms of Displacement Potential Function

In absence of body forces, the equilibrium equations for two dimensional plane stress/ plane strain elastic problems in terms of displacements components [33] are as follows

$$\frac{\partial^2 u}{\partial x^2} + \left(\frac{1-\mu}{2}\right)\frac{\partial^2 u}{\partial y^2} + \left(\frac{1+\mu}{2}\right)\frac{\partial^2 v}{\partial x.\partial y} = 0$$

$$\frac{\partial^2 v}{\partial y^2} + \left(\frac{1-\mu}{2}\right)\frac{\partial^2 v}{\partial x^2} + \left(\frac{1+\mu}{2}\right)\frac{\partial^2 u}{\partial x.\partial y} = 0 \qquad (2.22)$$

These two homogeneous elliptic partial differential equations with the appropriate boundary conditions should be sufficient for the evaluation of the two functions u and v, and the knowledge of these functions over the region concerned will uniquely determine the stress components.

Although the above two differential equations are sufficient to solve mixed boundary value elastic problems but in reality it is difficult to solve for two functions simultaneously. So, to overcome this difficulty, investigations are necessary to convert equations (Eq. 2.22) into a single equation of a single function. If that function is defined in terms of the displacement component u and v, then the determination of that function uniquely determines the stress functions sought for.

A potential function approach involves investigation of the existence of a function defined in terms of the displacement components. In this approach attempt had been made to reduce the problem to the determination of a single variable. A function ψ(x,y) is thus defined in terms of displacement components as,

$$u = \frac{\partial^2 \psi}{\partial x.\partial y}$$

$$v = -\left[\left(\frac{1-\mu}{1+\mu}\right)\frac{\partial^2\psi}{\partial y^2} + \left(\frac{2}{1+\mu}\right)\frac{\partial^2\psi}{\partial x^2}\right] \tag{2.23}$$

with this definition of ψ(x,y), the first of the two equations (Eq. 2.22) is automatically satisfied. Therefore, ψ has only to satisfy the second equation. Thus, the condition that ψ has to satisfy is

$$\frac{\partial^4\psi}{\partial x^4} + 2\frac{\partial^4\psi}{\partial x^2.\partial y^2} + \frac{\partial^4\psi}{\partial y^4} = 0 \tag{2.24}$$

Therefore, the problem is reduced to the evaluation of a single variable ψ(x,y) from the above bi-harmonic partial differential equation.

2.10 Boundary Conditions for the Function ψ for Mixed Boundary Value Problems

In order to solve the problem by solving for the function ψ of the bi-harmonic equation (Eq. 2.24), the boundary conditions should be expressed in terms of ψ. The boundary conditions are known restraints and loadings, that is, known values of components of stresses and displacements at the boundary. The relation between known functions and the potential function ψ at the boundary are [10]

$$u = \frac{\partial^2\psi}{\partial x.\partial y}$$

$$v = -\left[\left(\frac{1-\mu}{1+\mu}\right)\frac{\partial^2\psi}{\partial y^2} + \left(\frac{2}{1+\mu}\right)\frac{\partial^2\psi}{\partial x^2}\right] \tag{2.25}$$

$$\sigma_x = \frac{E}{(1+\mu)^2}\left[\frac{\partial^3\psi}{\partial x^2\partial y} - \mu\frac{\partial^3\psi}{\partial y^3}\right]$$

$$\sigma_y = -\frac{E}{(1+\mu)^2}\left[\frac{\partial^3\psi}{\partial y^3} + (2+\mu)\frac{\partial^3\psi}{\partial x^2\partial y}\right] \tag{2.26}$$

$$\sigma_{xy} = \frac{E}{(1+\mu)^2}\left[\mu\frac{\partial^3\psi}{\partial x^2\partial y} - \frac{\partial^3\psi}{\partial x^3}\right]$$

From the above expressions it is found that, as far as boundary conditions are concerned, either known restraints or known stresses or combinations of stresses and displacements, all can be converted to finite difference expressions in terms of ψ at the boundary.

Considering a pragmatic applicability, the rectangular components are converted into normal and tangential components, as these are actually known at the boundary using the following relationship (Eq. 2.15 and Eq. 2.16) [32].

$$u_n = u_x . l + u_y . m$$

$$u_t = u_y . l - u_x . m \tag{2.16}$$

$$\sigma_n = \sigma_{xx} . l^2 + \sigma_{yy} . m^2 + 2\sigma_{xy} . l\, m$$

$$\sigma_t = \sigma_{xy} . (l^2 - m^2) + (\sigma_{yy} - \sigma_{xx}) . l\, m \tag{2.15}$$

2.11 Treatment at the interfaces of three layer (sandwich-structure)

In the present study, casing materials are considered as perfectly bonded with the core material at the interface as it is considered molecular bonding between materials. The components of displacement (u and v) at the interfaces for the material layers are same. Hence, the displacement components u and v, defining the deformation are continuous over the three layers. It is interesting to note that in the case of constant body forces the compatibility equation (Eq. 2.19) determining stress distribution is independent of the elastic constants of the materials. In absence of body forces, the new displacement potential function approach yields a bi-harmonic partial differential equation (Eq. 2.24) from the equilibrium equations (Eq. 2.22). The obtained bi-harmonic equation is independent of the elastic constant such as poisson's ratio and modulus of elasticity of the materials, making it a continuity equation over the materials unless there is no separation at the interfaces of the materials. Therefore, the governing equation (Eq. 2.24) is valid at the inner points of the interface line of the layers as well as in the layers until any separation occurs at the interface. At the boundary points of the interface lines, the boundary conditions have the contribution to both the materials. Hence, special numerical technique is required to account the effects of boundary conditions on the three material layers simultaneously. This is discussed in chapter 3 of the dissertation.

2.12 Selection of Boundary Conditions

The possible known boundary components at a boundary point are any two out of four quantities, namely, u_n and u_t, the normal and tangential displacement components, σ_n and σ_t, the normal and tangential stress components. The possible sets of boundary conditions can be-

(i) Normal displacement component (u_n)
Tangential displacement component (u_t)

(ii) Normal displacement component (u_n)
Normal stress component (σ_n)

(iii) Normal displacement component (u_n)
Tangential stress component (σ_t)

(iv) Tangential displacement component (u_t)
Normal stress component (σ_n)

(v) Tangential displacement component (u_t)
Tangential stress component (σ_t)

(vi) Normal stress component (σ_n)
Tangential stress component (σ_t)

But among the above six sets of boundary conditions, sets (ii) and (v) do not usually occur in practical problems. So the remaining four possible sets of boundary conditions at any point on the boundary, which are considered in the present study are

1. (u_n, u_t)

2. (u_n, σ_t)

3. (u_t, σ_n)

4. (σ_n, σ_t)

CHAPTER 3

NUMERICAL MODEL

3.1 Introduction

Numerical models are mathematical models that utilize the numerical tools to obtain approximate solutions (not the exact solutions) of mathematical problems. In order to save computational time and cost, researchers are extensively using numerical methods to find solutions for the problems which are inconvenient to solve using analytical methods. For the plates of isotropic materials, analytical solutions for deflection, strain, and stress are available since 1898 [34]. But solving the problems of sandwich-structured composite by analytical method is still at preliminary stage. Therefore, the problems of sandwich-structured composite could not be solved analytically and it became necessary to solve the problem by experimental and numerical techniques. This study is based on the numerical techniques of solving the governing differential equations of a sandwich-structured composite and the process is described thereby.

The solution for the two-dimensional sandwich-structured composite comprises the solution of a fourth order partial differential equation with necessary boundary conditions, as stated in last chapters. Therefore, this chapter focuses on the solution of the problem by two well established numerical methods which are:

(i) Finite Difference method and
(ii) Finite Element method.

3.2 Finite Difference Method

The derivatives of a differential equation are replaced by the finite divided difference formulae (approximations of Taylor's series) for derivatives in finite difference technique. So a differential equation is converted into a set of linear algebraic equation which can be solved by a suitable technique. Since all finite difference formulae are approximation of infinite series of differences, it is necessary that the series should converge or the error caused by the truncation should be sufficiently small to give a reliable result.

In this method, the region of the body under consideration is divided by lines parallel to the co-ordinate axes. And points hence formed at the intersection of the these lines are treated as a grid of finite number of discrete points which are called node points as shown in Figure 3.1. The finite difference form of governing partial differential equation is applied to all node points except the boundary node points and that of appropriate boundary conditions are applied to the boundary node points. This gives a complete set of simultaneous equations, i.e. number of equations in the set is equal to the number of grid points, which is solved by a suitable numerical technique.

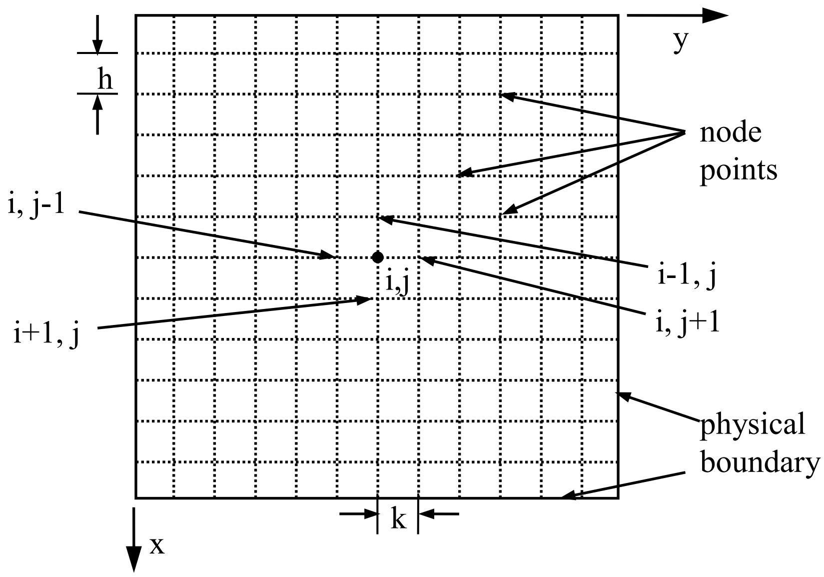


Figure 3.1: Discretization of rectangular body into a grid of points.

In the remaining portion of the chapter, the conversion procedure of the partial differential equation (2.24) and boundary conditions (2.25 and 2.26) in the form of difference equations is provided.

3.2.1 Application Technique of Finite Difference Formulae in Rectangular Grid

Usually in the region of study, where the dependent function ($\psi(x, y)$) has to be evaluated, the governing differential equation (Eq. 2.24) is applied at all node points except the boundary node points and boundary conditions (Eq. 2.25 and 2.26) are applied at boundary node points. For a rectangular shaped body usually two boundary conditions are known in each side of the rectangle. If a very simple problem of a sandwich-structured cantilever beam (Figure 3.2a) is considered then boundary conditions are shown in Figure 3.2c. So each side has two boundary conditions and if this body is transformed into a grid of discrete points then can be shown by Figure 3.2c.

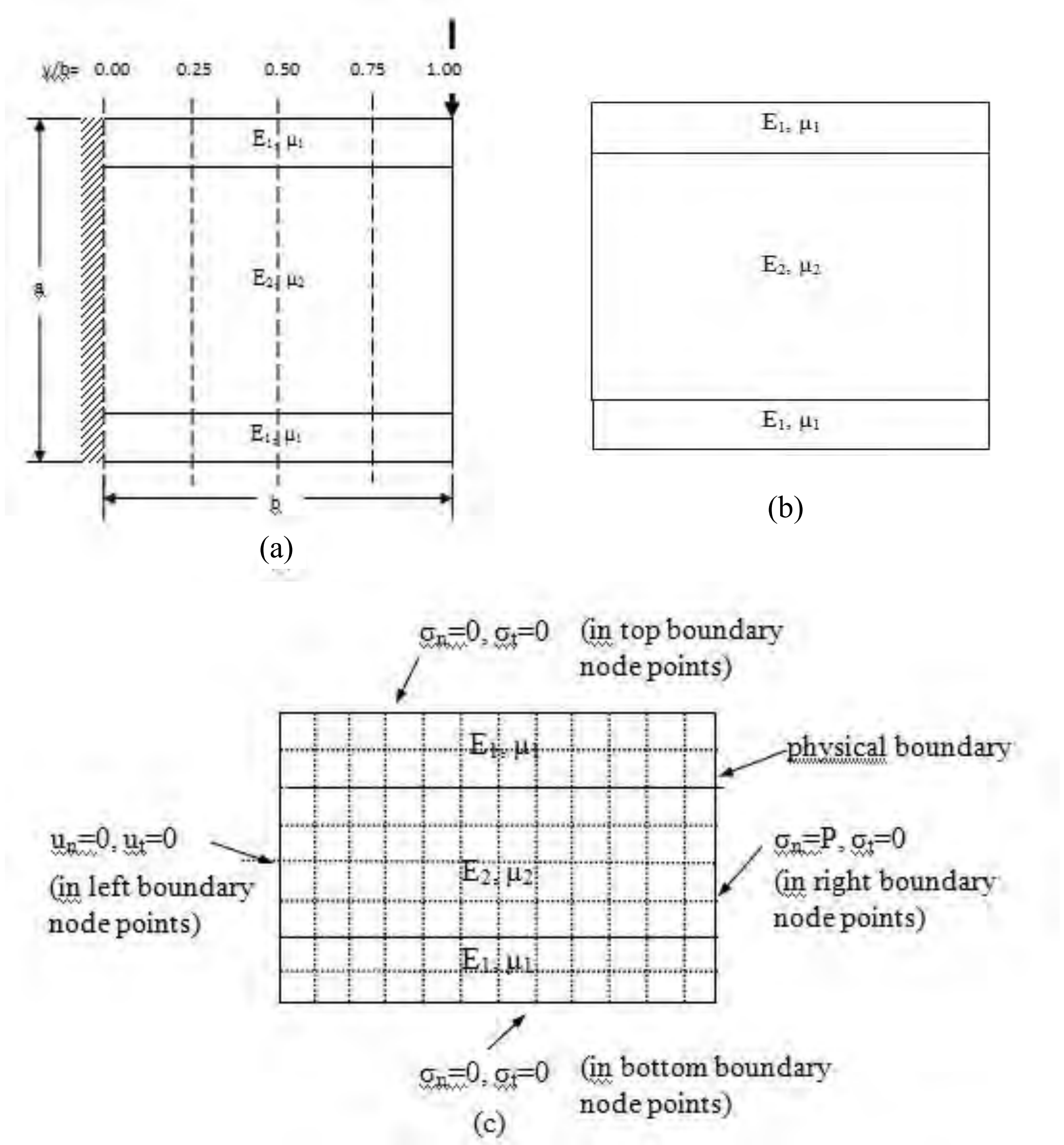


Figure 3.2: Boundary conditions for a cantilever beam

To overcome this problem a boundary near the physical boundary is assumed to exist which is named as imaginary boundary. If only top boundary is considered then it can be shown by Figure 3.3b. Top boundary nodes have two boundary conditions to satisfy, i.e. σ_n=0, σ_t=0. Hence an imaginary boundary is assumed at the outside of top physical boundary, immediate top grid points of the top boundary node points, as well as at all other boundaries of the rectangle. Both the boundary conditions σ_n=0 and σ_t=0 are satisfied at the inner physical boundary but for assigning the equations for each node (physical or imaginary) one of them is assigned for the physical node and the other one is assigned at the imaginary node, or vice versa. So the system of linear equations will have same number of variables and equations. In this work this technique is followed.

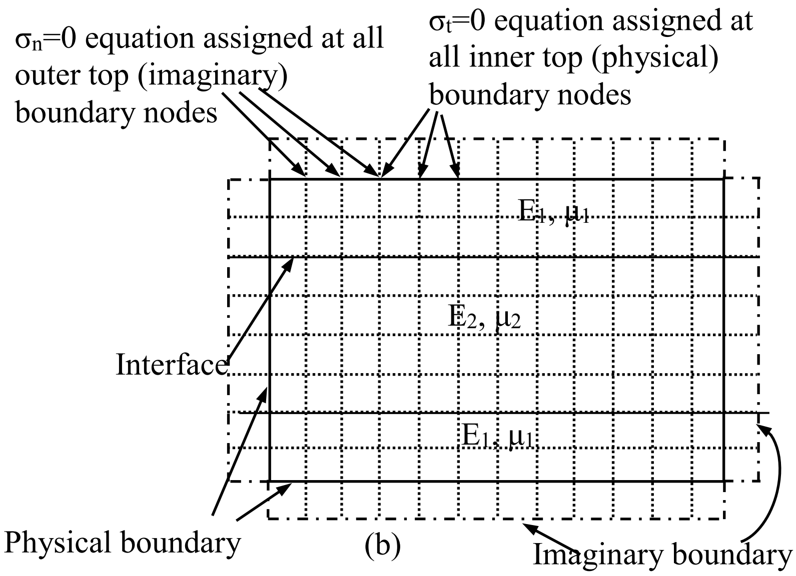


Figure 3.3: Boundary condition management with imaginary boundary

So the difference equations have to develop in such a way that they would cover the physical boundary points, inner points and imaginary points also. These finite difference forms are described in the following sections.

3.2.2 Finite Difference Form of the Bi-harmonic Governing Equation

The governing equation in terms of displacement potential function can be written as, from Eq. 2.24, [10]

$$\frac{\partial^4\psi}{\partial x^4} + 2\frac{\partial^4\psi}{\partial x^2.\partial y^2} + \frac{\partial^4\psi}{\partial y^4} = 0$$

By using the difference formula of $\partial^4 f/\partial x^4$, $\partial^4 f/\partial x^2 \partial y^2$ and $\partial^4 f/\partial y^4$, the above equation can be written as

$$zk1\{\psi(i-2,j)+\psi(i+2,j)\} - zk2\{\psi(i-1,j)+\psi(i+1,j)\} - zk3\{\psi(i,j+1)+\psi(i,j-1)\} + zk4.\psi(i,j) + zk5\{\psi(i-1,j-1)+\psi(i-1,j+1)+\psi(i+1,j-1)+\psi(i+1,j+1)+\psi(i,j-2)+\psi(i,j+2)\} = 0 \tag{3.1}$$

where, $zk1 = r^4$

$zk2 = 4(r^4 + r^2)$

$zk3 = 4(1 + r^2)$

$zk4 = (6r^4 + 8r^2 + 6)$

$zk5 = 2r^2$

The above equation (Eq. 3.1) is the finite difference approximation of the bi-harmonic partial differential equation and valid for all inner node points of the region i.e. all points of the region except the boundary points. The stencil of this equation is shown in Figure 3.4.

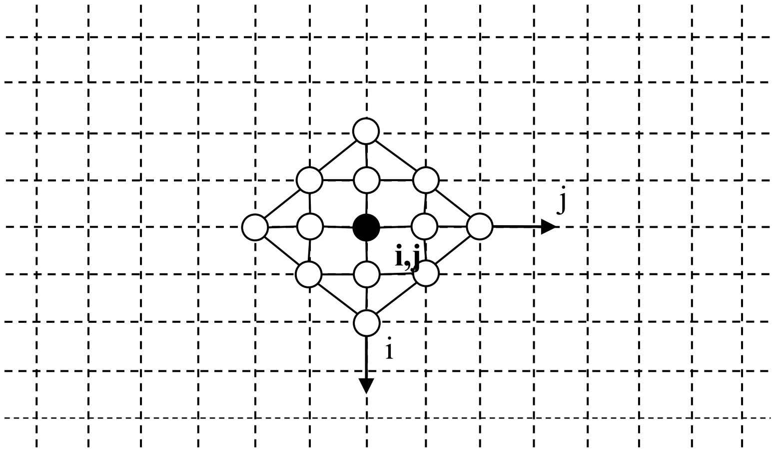


Figure 3.4: Stencil Arrangement of the governing equation

3.2.3 Finite Difference Form of the Boundary Conditions

For the application of boundary conditions the whole region is divided into five sectors, mentioned earlier in this chapter, which are i) top-left, ii) bottom-left, iii) bottom-right, iv) top-right, and v) boundary region associated with the interface. So a combination of forward, backward and central finite difference forms are required for different sections, such as for top-left boundary it requires both i-forward, j-forward difference equations.

3.2.4.1 Top-Left

The expressions of boundary conditions for top-left section (A-B-C section in Figure 3.8) are stated below. Boundary conditions in terms of displacement potential function can be written from equation (Eq. 2.25).

$$u_x(i,j) = \left(\frac{\partial^2\psi}{\partial x.\partial y}\right)_{i,j}$$

$$= s1.[9\psi(i,j) - 12\{\psi(i,j+1) + \psi(i+1,j)\} + 16\psi(i+1,j+1) + \psi(i+2,j+2) + 3\{\psi(i,j+2) + \psi(i+2,j)\} - 4\{\psi(i+1,j+2) + \psi(i+2,j+1)\}] \quad (3.2)$$

$$u_y(i,j) = -\left[\left(\frac{1-\mu}{1+\mu}\right)\frac{\partial^2\psi}{\partial y^2} + \left(\frac{2}{1+\mu}\right)\frac{\partial^2\psi}{\partial x^2}\right]$$

$$= s2.[zk7\{\psi(i-1,j) + \psi(i+1,j)\} + zk6\{\psi(i,j+1) + \psi(i,j-1)\} - \psi(i,j)] \quad (3.3)$$

$$\frac{\sigma_x}{E} = \frac{1}{(1+\mu)^2}\left[\frac{\partial^3\psi}{\partial x^2\,\partial y} - \mu\frac{\partial^3\psi}{\partial y^3}\right]$$

$$= s3[1.5\psi(i,j-1) + (6zk9-5)\psi(i,j) + (6-8zk9)\psi(i,j+1) + (2zk9-3)\psi(i,j+2) + 0.5\psi(i,j+3) - 3zk9\{\psi(i-1,j) + \psi(i+1,j)\} + 4zk9\{\psi(i-1,j+1) + \psi(i+1,j+1)\} - zk9\{\psi(i-1,j+2) + \psi(i+1,j+2)\}] \quad (3.4)$$

$$\frac{\sigma_y}{E} = -\frac{1}{(1+\mu)^2}\left[\frac{\partial^3\psi}{\partial y^3} + (2+\mu)\frac{\partial^3\psi}{\partial x^2\,\partial y}\right]$$

$$= s4[1.5\psi(i,j-1) + (6zk10-5)\psi(i,j) + (6-8zk10)\psi(i,j+1) + (2zk10-3)\psi(i,j+2) + 0.5\psi(i,j+3) - 3zk10\{\psi(i-1,j) + \psi(i+1,j)\} + 4zk10\{\psi(i-1,j+1) + \psi(i+1,j+1)\} - zk10\{\psi(i-1,j+2) + \psi(i+1,j+2)\}] \quad (3.5)$$

$$\frac{\sigma_{xy}}{E} = \frac{1}{(1+\mu)^2}\left[\mu\frac{\partial^3\psi}{\partial x^2\,\partial y} - \frac{\partial^3\psi}{\partial x^3}\right]$$

$$= r.s3[-1.5\{\psi(i,j-1)+\psi(i,j+1)\} + (3-10zk9)\psi(i,j) + 3zk9\psi(i-1,j) + 2\{\psi(i+1,j-1)+\psi(i+1,j+1)\} - 0.5\{\psi(i+2,j-1) + \psi(i+2,j+1)\} + (12zk9-4)\psi(i+1,j) + (1-6zk9)\psi(i+2,j) + zk9\psi(i+3,j)] \quad (3.6)$$

Where,

$$s1 = \frac{1}{4rh^2}\;; \qquad s2 = -\frac{1}{zk8}\;; \qquad s3 = -\frac{1}{p2}\;; \qquad s4 = -\frac{1}{\mu_i p2}$$

$$p2 = \frac{(1+\mu_i)^2 r^3 h^3}{\mu_i}; \qquad zk6 = \frac{(1-\mu_i)}{2(1-\mu_i+2r^2)}\;; \qquad zk7 = \frac{r^2}{2(1-\mu_i+2r^2)}\;;$$

$$zk8 = \frac{(1+\mu_i)r^2h^2}{2(1-\mu_i+2r^2)}\;; \qquad zk9 = \frac{r^2}{2\mu_i}\;; \qquad zk10 = \frac{r^2(1+\mu_i)}{2}$$

i=1 for the upper portion of the interface, i=2 for the lower portion of the interface The programming scheme is used in such a way that there is a single parameter in the formula but the constants are subsequently changed depending on the points located in upper or lower portion of the interface.

For the application of the formulae of the boundary conditions, only the single set of formulae for top-left (A-B-C) can be applied at the other boundary sections such as bottom-left (D-E-F), top-right (A-J-I) and bottom-right (F-G-H) as shown in Figure 3.5 incorporating the replacement of some constants mentioned in Table 3.1.

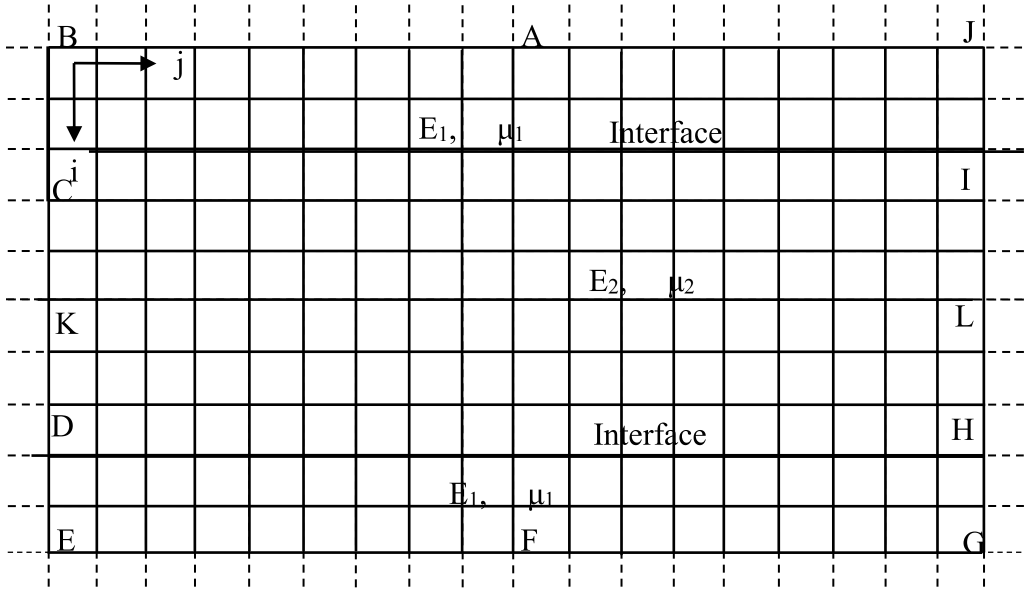


Figure 3.5: Different sections in a specified problem

Table 3.1: Formula conversion at the different sections of the boundary of the sandwich structured composite

Boundary region	Type of difference formula required	Replacement required
Top-left	i-forward j-forward	No change required
Bottom-left	i-backward j-forward	r by –r, s1 by –s1, i-* by i+* and i+* by i-*
Bottom-right	i-backward j- backward	s3 by –s3, s4 by –s4, i-* by i+*, i+* by i-*, j-* by j+*, and j+* by j-*
Top-right	i- backward, j- forward	r by –r, s3 by –s3, s4 by –s4, i-* by i+*, i+* by i-*, j-* by j+*, and j+* by j-*

Where, * stands for digits 1 or 2 or 3 etc.

The stencils of the equations of the boundary conditions are illustrated in Figure 3.6 and 3.7.

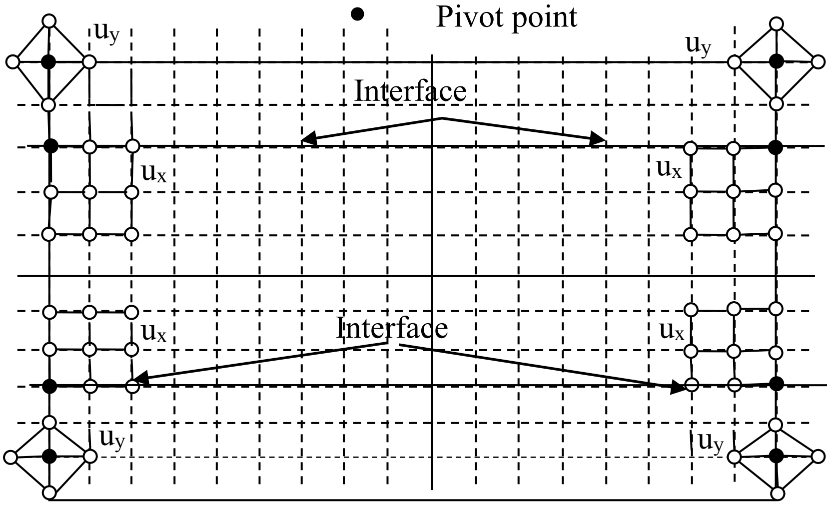


Figure 3.6: Stencils of u_x and u_y for different region of the boundary

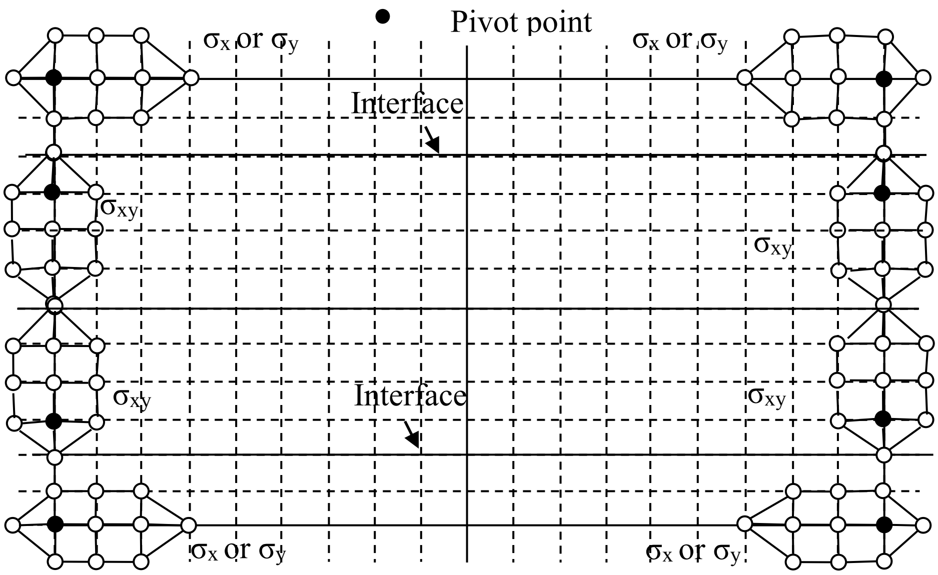


Figure 3.7: Stencils of σ_x , σ_y and σ_{xy} for different region of any two layer of the boundary

3.2.4.2 Interface-Left

At the interface, different combinations of formula structures are trialed, but the successful one is illustrated in the next section of the dissertation. The displacement component u_x is independent of the material properties and continuous over the sandwich structured composite. Thus, the formulation of u_x remains same as the earlier section 3.2.4.1, 3.2.4.2, 3.2.4.3, and 3.2.4.4.

$$u_x(i,j) = \left(\frac{\partial^2 \psi}{\partial x . \partial y}\right)_{i,j}$$

$$= s1.[9\psi(i,j) - 12\{\psi(i,j+1) + \psi(i+1,j)\} + 16\psi(i+1,j+1) + \psi(i+2,j+2) + 3\{\psi(i,j+2) + \psi(i+2,j)\} - 4\{\psi(i+1,j+2) + \psi(i+2,j+1)\}] \quad (3.41)$$

The displacement component u_y is dependent on the material properties and continuous over the layers of sandwich-structured composite. At the interface, two materials are perfectly bonded together, hence the displacement component u_y of the common node point is the average of the two displacement components u_{y1} and u_{y2} considering through the each side of the material with average of poisons ratios. So, at the left side of the upper interface it could be written as

$$u_{ylu} = -\left[\left(\frac{1-\mu}{1+\mu}\right)\frac{\partial^2 \psi}{\partial y^2} + \left(\frac{2}{1+\mu}\right)\frac{\partial^2 \psi}{\partial x^2}\right]$$

$$= s2u.[e71\{\psi(i-1,j) + \psi(i+1,j)\} + e61\{\psi(i,j+1) + \psi(i,j-1)\} - \psi(i,j)]$$

(3.7)

Where,

$$s2u = -\frac{1}{e81};$$

$$e61 = \frac{(1 - (\mu_1 + \mu_2)/2)}{2[1 - \{(\mu_1 + \mu_2)/2\} + 2r^2]}; \qquad e71 = \frac{r^2}{2[1 - \{(\mu_1 + \mu_2)/2\} + 2r^2]};$$

$$e81 = \frac{(1 + (\mu_1 + \mu_2)/2)r^2h^2}{2[1 - \{(\mu_1 + \mu_2)/2\} + 2r^2]}$$

And, at the left side of the lower interface it could be written as

$$u_{yll} = -\left[\left(\frac{1-\mu}{1+\mu}\right)\frac{\partial^2\psi}{\partial y^2} + \left(\frac{2}{1+\mu}\right)\frac{\partial^2\psi}{\partial x^2}\right]$$

$$= s2l.\,[e72\{\psi(i-1,j) + \psi(i+1,j)\} + e62\{\psi(i,j+1) + \psi(i,j-1)\} - \psi(i,j)]$$

(3.8)

Where,

$$s2l = -\frac{1}{e82}\ ;$$

$$e62 = \frac{(1-(\mu_2+\mu_3)/2)}{2[1-\{(\mu_2+\mu_3)/2\}+2r^2]}\ ; \qquad e72 = \frac{r^2}{2[1-\{(\mu_2+\mu_3)/2\}+2r^2]}\ ;$$

$$e82 = \frac{(1+(\mu_2+\mu_3)/2)r^2h^2}{2[1-\{(\mu_2+\mu_3)/2\}+2r^2]}$$

The stencil of the displacement component is illustrated in the following Figure 3.8.

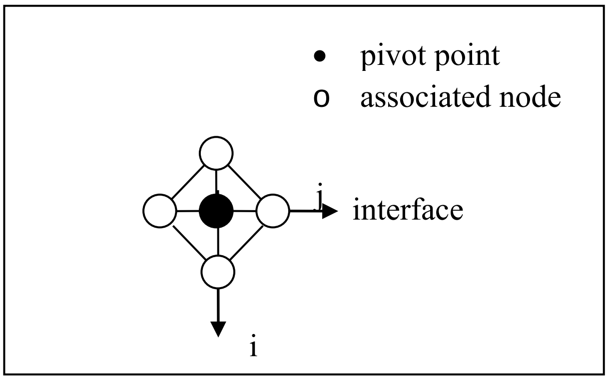


Figure 3.8: Stencil of u_y at the left points of two interface line

3.2.4.3 Interface-Right

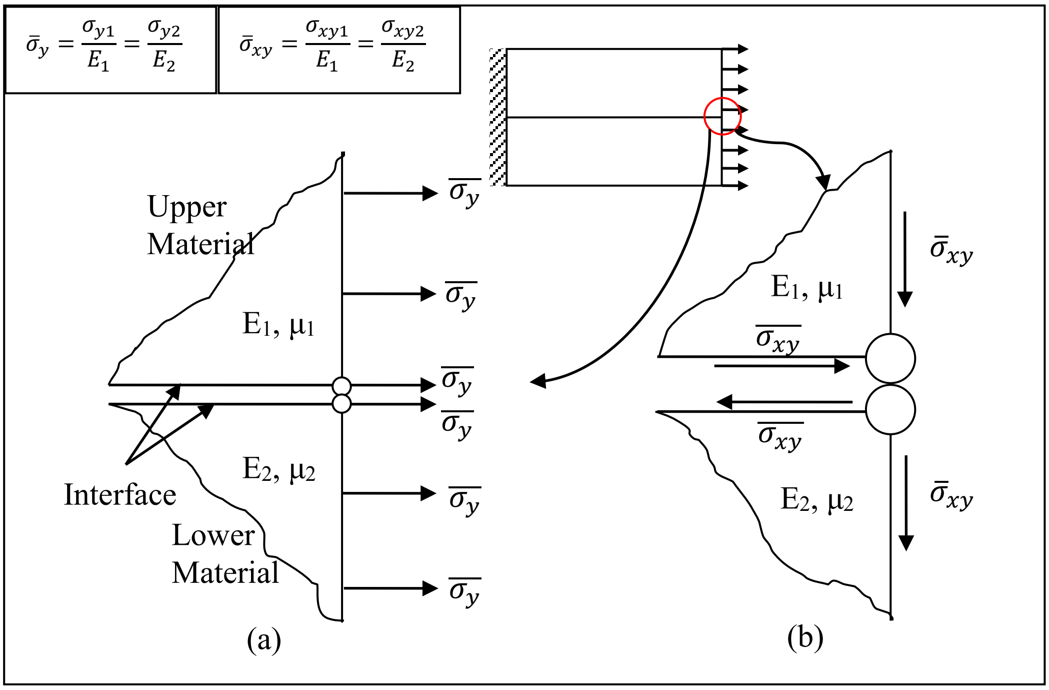


Figure 3.9: At the right interface point (a) Normal stress (b) Shear stress

At the interface two points from upper and lower material are actually bonded together in each of the two layers of a sandwich-structured composite beam. The normal stress acting at the interface boundary point is shown in Figure 3.9 (a).

The average of normal stresses based on the displacement potential function of upper material and that of lower material should be equal to the applied normal stress at that point. Thus for the upper right interface it could be written as-

$$\frac{\sigma_{yru}}{E} = -\frac{1}{(1+\mu)^2}\left[\frac{\partial^3\psi}{\partial y^3} + (2+\mu)\frac{\partial^3\psi}{\partial x^2\,\partial y}\right]$$

$$= s4u[1.5\psi(i,j-1) + (6e10-5)\psi(i,j) + (6-8e10)\psi(i,j+1) + (2e10-3)\psi(i,j+2) + 0.5\psi(i,j+3) - 3e10\{\psi(i-1,j) + \psi(i+1,j)\} + 4e10\{\psi(i-1,j+1) + \psi(i+1,j+1)\} - e10\{\psi(i-1,j+2) + \psi(i+1,j+2)\}] \quad (3.9)$$

Where,

$$s4u = -\frac{2}{(\mu_1 + \mu_2)p2}; \qquad e10 = \frac{r^2\{1 + (\mu_1 + \mu_2)/2\}}{2}$$

And for the lower right interface it could be written as-

$$\frac{\sigma_{yrl}}{E} = -\frac{1}{(1+\mu)^2}\left[\frac{\partial^3\psi}{\partial y^3} + (2+\mu)\frac{\partial^3\psi}{\partial x^2\,\partial y}\right]$$

$$= s4l[1.5\psi(i,j-1) + (6e11-5)\psi(i,j) + (6-8e11)\psi(i,j+1) + (2e11-3)\psi(i,j+2) + 0.5\psi(i,j+3) - 3e11\{\psi(i-1,j) + \psi(i+1,j)\} + 4e11\{\psi(i-1,j+1) + \psi(i+1,j+1)\} - e11\{\psi(i-1,j+2) + \psi(i+1,j+2)\}] \qquad (3.10)$$

Where,

$$s4l = -\frac{2}{(\mu_1 + \mu_2)p2}; \qquad e11 = \frac{r^2\{1 + (\mu_2 + \mu_3)/2\}}{2}$$

The stencil of the normal stress at the interface is shown in Figure 3.13(a).

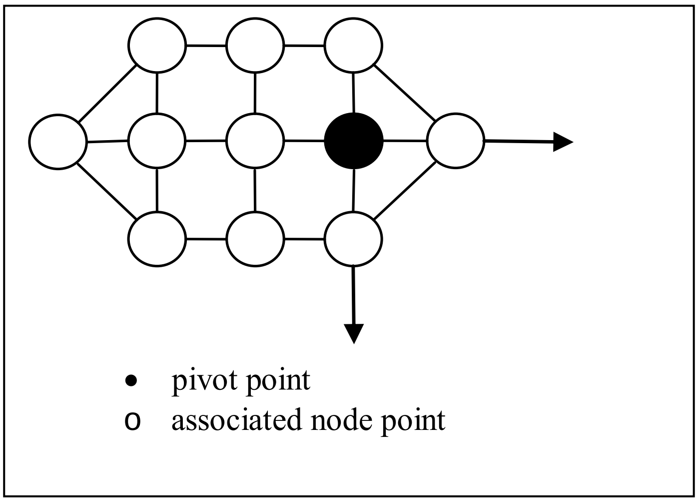


Figure 3.10: Stencil of the (a) normal stress (b) tangential stress at the right point of the interface line

The tangential stress acting at the interface boundary point is shown in Figure 3.10 (b). According to the sign convention, shear stress causes anticlockwise moment to the body is considered as positive. The positive shear stresses are shown in Figure 3.10 (b) at the interfaces of the layers of sandwich-structured composite. The average of shear stresses based on the displacement potential function of upper material and that of lower material should be equal to the applied shear stress at that point. Thus for right side of upper interface it could be written as-

$$\frac{\sigma_{xyru}}{E} = \frac{1}{(1+\mu)^2}\left[\mu\frac{\partial^3\psi}{\partial x^2\,\partial y} - \frac{\partial^3\psi}{\partial x^3}\right]$$

$$= r.\,s3u[-1.5\{\psi(i,j-1) + \psi(i,j+1)\} + (3 - 10e8)\psi(i,j) + 3e8\psi(i-1,j) + 2\{\psi(i+1,j-1) + \psi(i+1,j+1)\} - 0.5\{\psi(i+2,j-1) + \psi(i+2,j+1)\} + (12e8 - 4)\psi(i+1,j) + (1 - 6e8)\psi(i+2,j) + e8\psi(i+3,j)] \qquad (3.11)$$

Where,

$$s3u = -\frac{1}{p2}\,; \qquad e8 = \frac{r^2}{\mu_1 + \mu_2}$$

And for right side of lower interface it could be written as-

$$\frac{\sigma_{xyrl}}{E} = \frac{1}{(1+\mu)^2}\left[\mu\frac{\partial^3\psi}{\partial x^2\,\partial y} - \frac{\partial^3\psi}{\partial x^3}\right]$$

$$= r.\,s3l[-1.5\{\psi(i,j-1) + \psi(i,j+1)\} + (3 - 10e9)\psi(i,j) + 3e9\psi(i-1,j) + 2\{\psi(i+1,j-1) + \psi(i+1,j+1)\} - 0.5\{\psi(i+2,j-1) + \psi(i+2,j+1)\} + (12e9 - 4)\psi(i+1,j) + (1 - 6e9)\psi(i+2,j) + e9\psi(i+3,j)] \qquad (3.12)$$

Where,

$$s3l = -\frac{1}{p2}\,; \qquad e9 = \frac{r^2}{\mu_2 + \mu_3}$$

3.2.5 Evaluation of ψ

If the whole region is placed in a rectangular grid then the region gives a finite number of node points which include reference boundary points, imaginary boundary points and inner body points (node points other than the reference and imaginary boundary points). Finite difference expressions of the boundary conditions should be applied in all reference boundary and imaginary boundary node points. And Finite difference expressions of the governing equation should be applied in all inner body points. So every point gives rise to a linear algebraic equation and the whole region gives a set of linear algebraic equations equal to the number of total node points in the region. The set of linear equations can be shown as,

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & . & . & . & a_{1n} \\ a_{21} & a_{22} & a_{23} & . & . & . & a_{2n} \\ a_{31} & a_{32} & a_{33} & . & . & . & a_{3n} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ a_{n1} & a_{n2} & a_{n3} & . & . & . & a_{nn} \end{bmatrix} \begin{Bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ . \\ . \\ \psi_n \end{Bmatrix} = \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \\ . \\ . \\ c_n \end{Bmatrix} \tag{3.13}$$

or

$$[A]\{\psi\} = \{C\}$$

where, a_{11}, a_{12} ……a_{nn} are coefficients, n is the number of total node points, [A] is called coefficient matrix and [C] is constant matrix.

In this equation only unknowns are the ψ's. Many numerical techniques are available to solve this type of equation such as L-U decomposition, cholesky composition, gauss-siedel method, matrix portioning, matrix inversion, relaxation method etc. Here L-U decomposition method is used and hence value of ψ at each node point will be found.

3.2.6 Determination of Stress and Displacement Component at Each Grid Point

Once value of ψ at every node points are evaluated the stress and displacement components at each point can be found from the equations (Eq. 2.25and 2.26).

In order to calculate stress and displacement the finite difference expressions of these equations are required and as before the expressions depend on the section of the region where these should be applied. For bottom-left section the finite difference equations are given by equations 3.2 to 3.6.

These equations can also be used for the other sections of the region except at the interface by changing signs of the constants i, j, s1, s2, s3, and s4 as shown in Table 3.2. The formula structures to calculate the stresses and displacements in the different sections (Figure 3.11) of the sandwich-structured composite are shown in Figure 3.12, and 3.14.

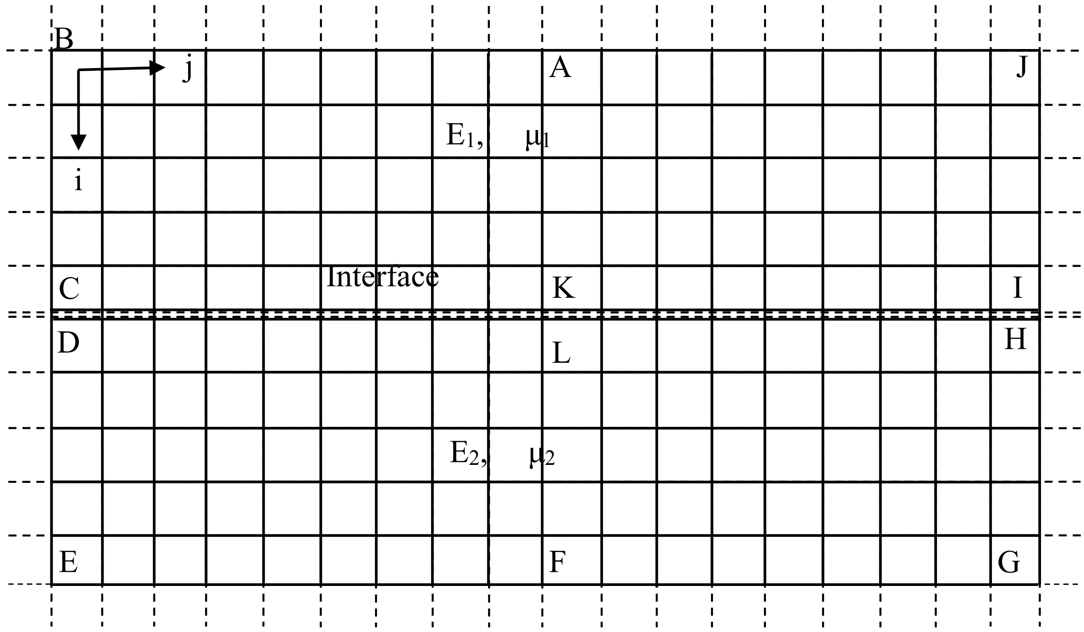


Figure 3.11: Different sections of a two dimensional rectangular body

Table 3.2: Formula conversion at the different sections of the body

Boundary section of the region	Type of difference formula required	Replacement required
Top-left (ABCK)	i-forward j-forward	No change required
Bottom-left (DEFL)	i-backward j-forward	r by –r, s1 by –s1, i-* by i+* and i+* by i-*
Bottom-right (LFGH)	i-backward j- backward	s3 by –s3, s4 by –s4, i-* by i+*, i+* by i-*, j-* by j+*, and j+* by j-*
Top-right (IJAK)	i- backward, j- forward	r by –r, s3 by –s3, s4 by –s4, i-* by i+*, i+* by i-*, j-* by j+*, and j+* by j-*

Where, * stands for digits 1 or 2 or 3 etc.

At the interface, special formulations are required for the calculation of stress and displacement. The formulation is derived in such a way that no nodal point of the formulation of one material lies to the region of another material. The formula structures to calculate the stresses and displacements at the interface line of the composite are shown in Figure 3.13, and 3.15. Although there are two points at the interface, the displacement components u_x and u_y are same for the two points, as the two points at the interface are perfectly bonded together. But the stress components vary at the two points as the mechanical properties are different for upper and lower portion of the interface.

For left side of the interface line for upper portion of the sandwich-structured composite-

$$u_x(i,j) = \left(\frac{\partial^2 \psi}{\partial x.\partial y}\right)_{i,j}$$

$$= s1.[-9\psi(i,j) + 12\{\psi(i,j-1) + \psi(i-1,j)\} - 16\psi(i-1,j-1) + \psi(i-2,j-2) - 3\{\psi(i,j-2) + \psi(i-2,j)\} + 4\{\psi(i-1,j-2) + \psi(i-2,j-1)\}] \tag{3.14}$$

$$u_y = \frac{1}{2}\left\{u_{y1})_{i,j} + u_{y2})_{i,j}\right\}$$

$$u_y = \frac{1}{2}\{(-2c21 + 2c31 - 2c22 + 2c32)\psi(i,j) + (c21 + c22)\psi(i,j+1) + (c21 + c22)\psi(i-1,j) - 5c31.\psi(i-1,j) + 4c31.\psi(i-2,j) - c31.\psi(i-3,j) - 5c32.\psi(i+1,j) + 4c32.\psi(i+2,j) - c32.\psi(i+3,j)\} \tag{3.15}$$

$$\frac{\sigma_{x1}}{E_1} = \frac{1}{(1+\mu_1)^2}\left[\frac{\partial^3\psi}{\partial x^2\,\partial y} - \mu_1\frac{\partial^3\psi}{\partial y^3}\right]$$

$$\frac{\sigma_{x1}}{E_1} = 3c51.\psi(i,j-1) + (-6c41 - 10c51).\psi(i,j) + (8c41 + 12c51).\psi(i,j+1) + (-2c51 - 6c51).\psi(i,j+2) + c51.\psi(i,j+3) + 15c41.\psi(i-1,j) - 12c41.\psi(i-2,j) + 3c41.\psi(i-3,j) - 20c41.\psi(i-1,j+1) + 16c41.\psi(i-2,j+1) - 4c41.\psi(i-3,j+1) + 5c41.\psi(i-1,j+2) - 4c41.\psi(i-2,j+2) + c41.\psi(i-3,j+2) \tag{3.16}$$

Where,

$$c41 = \frac{1}{2rh^3(1+\mu_1)^2}\ ; \qquad c51 = \frac{\mu_1}{2r^3h^3.(1+\mu_1)^2}$$

$$\frac{\sigma_{y1}}{E_1} = -\frac{1}{(1+\mu_1)^2}\left[\frac{\partial^3\psi}{\partial y^3} + (2+\mu_1)\frac{\partial^3\psi}{\partial x^2\,\partial y}\right]$$

$$\frac{\sigma_{y1}}{E_1} = 3c71.\psi(i,j-1) + (6c61 - 10c71).\psi(i,j) + (-8c61 + 12c71).\psi(i,j+1) + (2c61 - 6c71).\psi(i,j+2) + c71.\psi(i,j+3) - 15c61.\psi(i-1,j) + 12c61.\psi(i-2,j) - 3c61.\psi(i-3,j) + 20c61.\psi(i-1,j+1) - 16c61.\psi(i-2,j+1) + 4c61.\psi(i-3,j+1) - 5c61.\psi(i-1,j+2) + 4c61.\psi(i-2,j+2) - c61.\psi(i-3,j+2) \tag{3.17}$$

$$\frac{\sigma_{xy1}}{E_1} = \frac{1}{(1+\mu_1)^2}\left[\mu_1\frac{\partial^3\psi}{\partial x^2\,\partial y} - \frac{\partial^3\psi}{\partial x^3}\right]$$

$$\frac{\sigma_{xy1}}{E_1} = 3c81.\psi(i, j-1) + d31.\psi(i,j) + 3c81.\psi(i, j+1) - 4c81\psi(i-1, j-1) + c81.\psi(i-2, j-1) + d41.\psi(i-1, j) + d51.\psi(i-2, j) + c91.\psi(i-3, j) - 4c81.\psi(i-1, j+1) + c81.\psi(i-2, j+1) \quad (3.18)$$

These equations can also be used for the other parts of the interface line by following the replacements of the constants as shown in Table 3.3.

3.2.7 Computer Program for the Finite Difference Solution

A computer program based on the FORTRAN language is developed for the finite difference solution of sandwich structured composite. The program has several subroutines to perform different tasks. In the flow chart as shown in Figure 3.16, the whole program is briefly expressed. It is actually a parsimonious representation of the whole program. First the program reads data from two input files. Input file 1 contains data about the region's shape specified in Cartesian coordinate, material properties, boundary conditions and the loading conditions. And input file 2 contains data about the interface position in the region, its shape expressed in Cartesian coordinate, interface boundary condition loading condition. It is mentionable that the input files have to prepare in a prescribed way, otherwise the program won't read and will show error message. Some typical input files and guideline to generate input files are shown in the Annexure A.

Table 3.3: Formula conversion at the different sections of the interface line of the sandwich-structured composite

Section at the interface	Type of difference formula required	Replacement required
Upper-left	i- backward j-forward	No change required
Lower-left	i- forward j-forward	E_1 by E_2, μ_1 by μ_2, i-* by i+*, i+* by i-*, c81 by -c82, d31 by –d31, d41 by –d42, d51 by –d52, c91 by –c92
Upper-right	i-backward j- backward	j-* by j+*, j+* by j-*, c41 by –c41, c51 by –c51, c61 by –c61, c71 by –c71
Lower-right	i- backward, j- forward	E_1 by E_2, μ_1 by μ_2, j-* by j+*, j+* by j-*, c41 by –c41, c51 by –c51, c61 by –c61, c71 by –c71

Where, * stands for digits 1 or 2 or 3 etc.

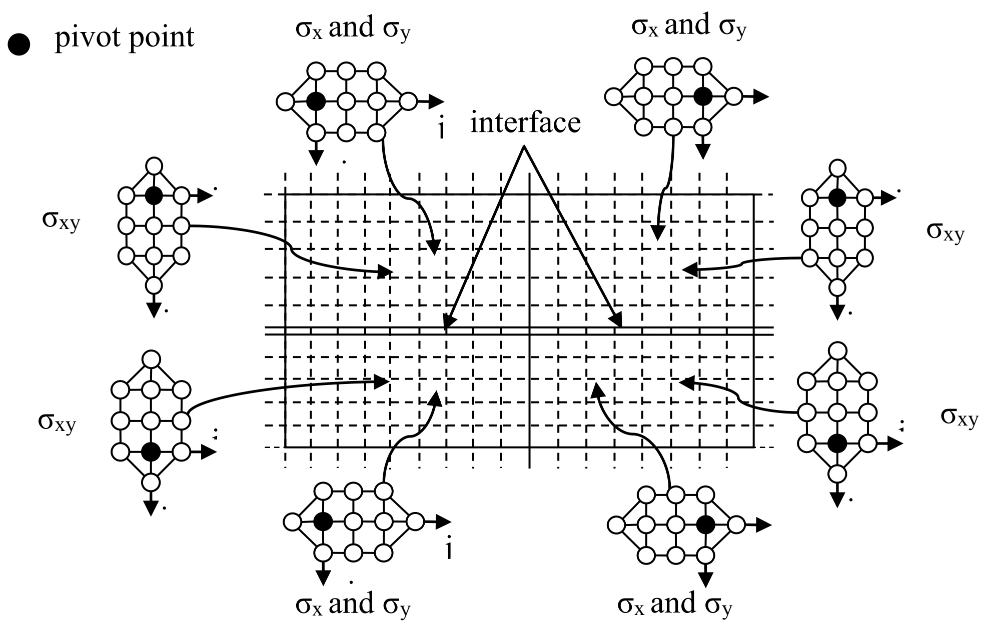


Figure 3.12: Different formula structures for stress calculation at different sections of the composite

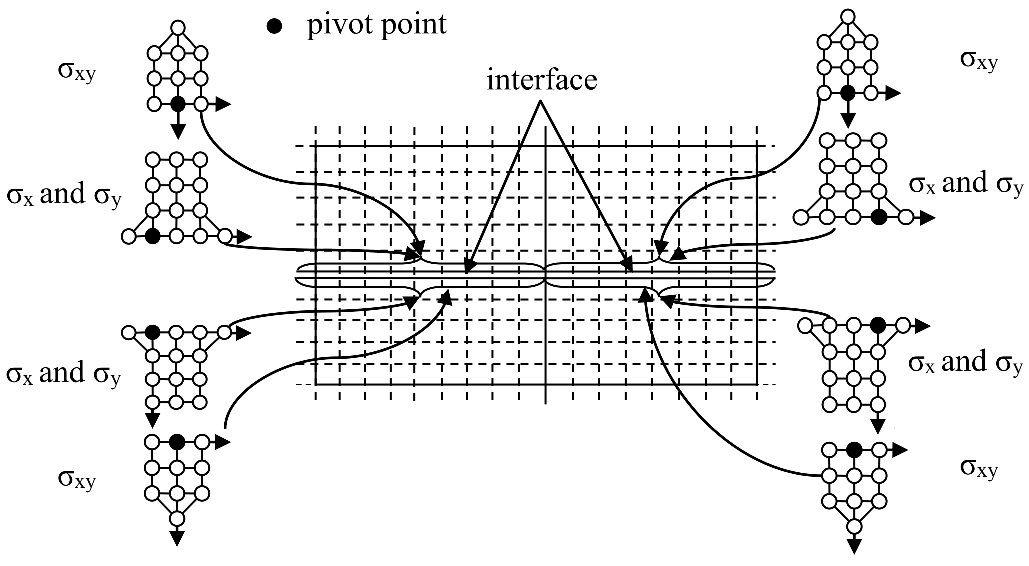


Figure 3.13: Different formula structures for stress calculation at the interfaces

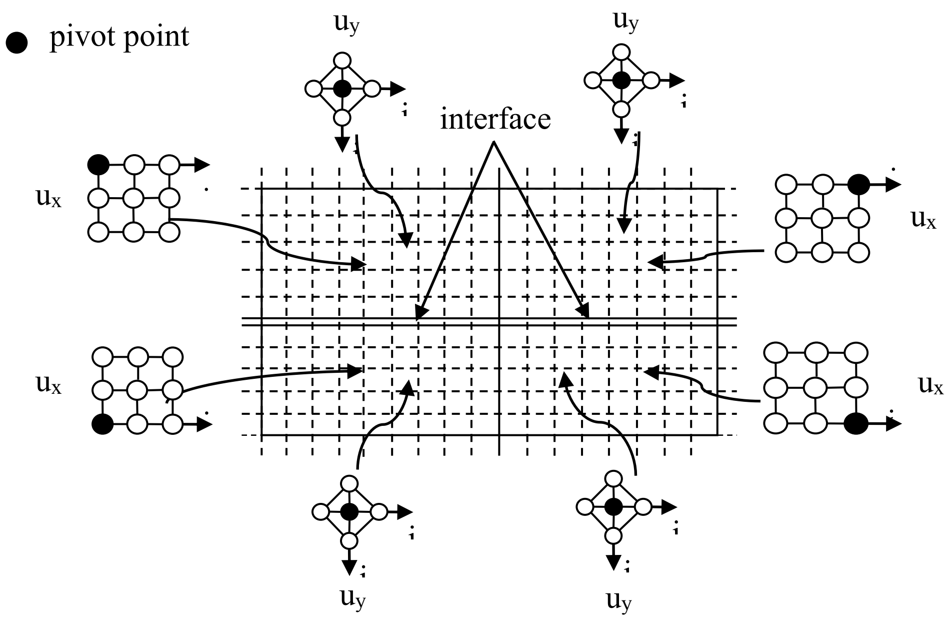


Figure 3.14: Different formula structures for displacement calculation at different sections of the composite

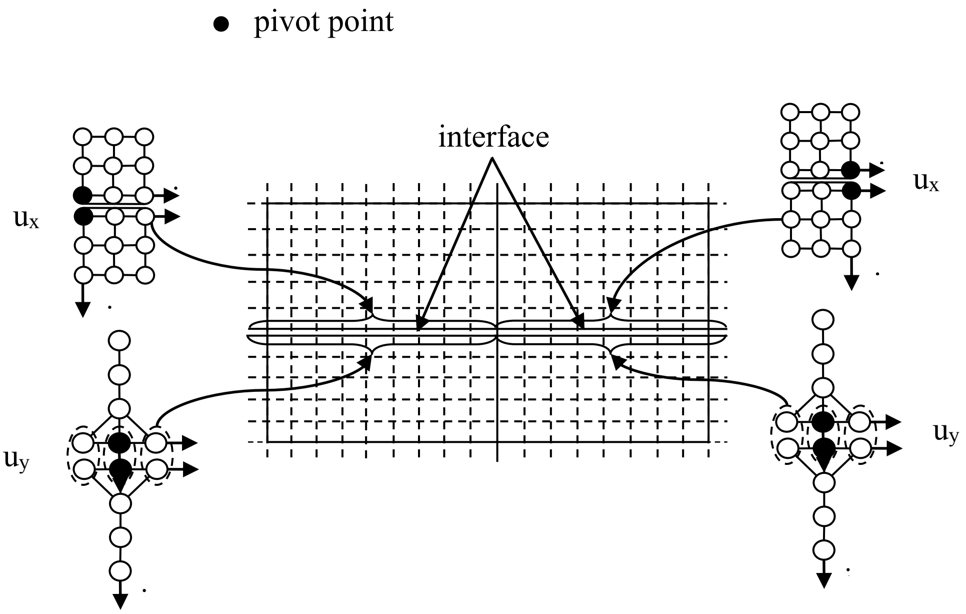


Figure 3.15: Different formula structures for displacement calculation at the interface

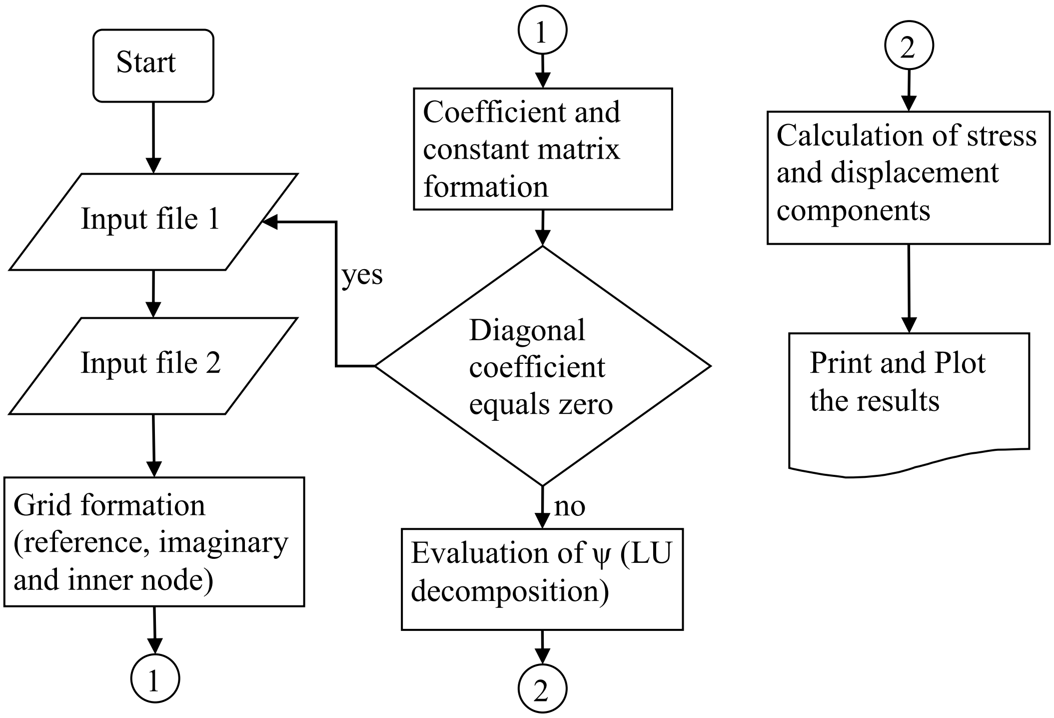


Figure 3.16: Flowchart of the computer program for finite difference solution

So the first subroutine reads these data from these input files and passes to the next subroutine which forms the grid. That means this subroutine divides the whole region into the rectangular grid as specified in the input file and hence makes the reference boundary, imaginary boundary and inner body node points as mentioned earlier in this chapter. It also does the node numbering and the printing of these numbering. Then the next subroutine generates the set of linear equations, i.e. the coefficient matrix and constant matrix by using the finite difference form of the boundary conditions (Eq. 3.41, 3.42, 3.43 and 3.44), as specified in the input files, and the governing equation (Eq. 3.40). For the solution of the set of equation it is necessary that the diagonal coefficients must be non zero. So if it happens that any one of the diagonal coefficient becomes zero then program will give error message and will stop automatically. To fix the problem the input files should be checked and then has to find the possible sources of error, such as error in specifying the boundary conditions or the coordinate of the region. So if everything is all right then the next subroutine decomposes the matrix and finds the solution of the only variable $\psi(x, y)$ in every node point. And after that the next subroutine calculates the stress and displacement components at every node point by using the finite difference equations stated before (Eq. 3.68, 3.77, 3.78. 3.79 and 3.80). Then three to four subroutines come to action for the printing and plotting of the result in a user defined way.

3.3 Finite Element Method

Finite element method is a very popular numerical method and used by many researchers over the world for solving a wide range of problems. In the previous method, finite difference method, the whole region is divided into a grid of discrete points or nodes and in each node finite difference form of the differential equation is applied which offers a point wise approximation. In contrast to the finite difference method, the finite element method divides the solution region into simply shaped regions or elements. An approximate solution for the differential equation is developed for each of these elements and the total solution is then generated by linking together the individual solutions to ensure the continuity at the inter-element boundary. So this technique provides piece wise approximation of the region rather than the point wise approximation found in the finite difference method. Based on finite element method several commercial software is available such as ANSYS, NASTRAN & PATRAN, FEMLAB, LS DIANA etc. which are very reliable and equally popular. In this study ANSYS is used to solve the problems and hence gives a way to compare and validate the finite difference results. Since finite difference solution is the main target of this work, finite element method here performs as a supporting tool. Therefore details of the solution procedures by commercial software ANSYS (finite element method) are not mentioned here.

However, brief description of the FEM modeling are depicted hereunder-

Details of FEM Modeling:

Preference: Structural

Element Type: Plane182- Solid- Quad 4 node 182

PLANE182 is used for 2-D modeling of solid structures. The element can be used as either a plane element (plane stress, plane strain or generalized plane strain) or an axisymmetric element. It is defined by four nodes having two degrees of freedom at each node: translations in the nodal x and y directions.

Material Model :

Material Model No 1: Linear Isotropic EXY=1.0, PRXY=0.29

Material Model No 2: Linear Isotropic EXY=0.9, PRXY=0.31

Modeling: Created by Key Points in Active Co-ordinate System

Geometry: a=1.00 unit and b=2.54054 unit

Key Points:

1(0,0); 2(2.54,0); 3(0, -0.16); 4(2.54,-0.16);

5(0,-0.68); 6(2.54,-0.68); 7(0,-1); 8(2.54,-1)

Element No: (47x37)=1739

Boundary Conditions: Left side Displacement is zero in all degree of freedom

Load assigned in specific nodes.

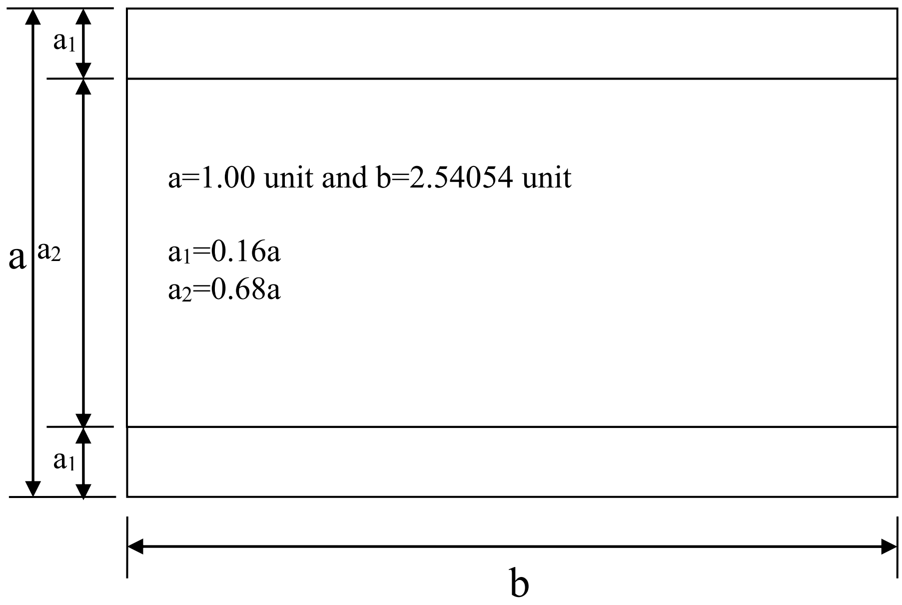


Figure 3.17: Geometry of FEM Model

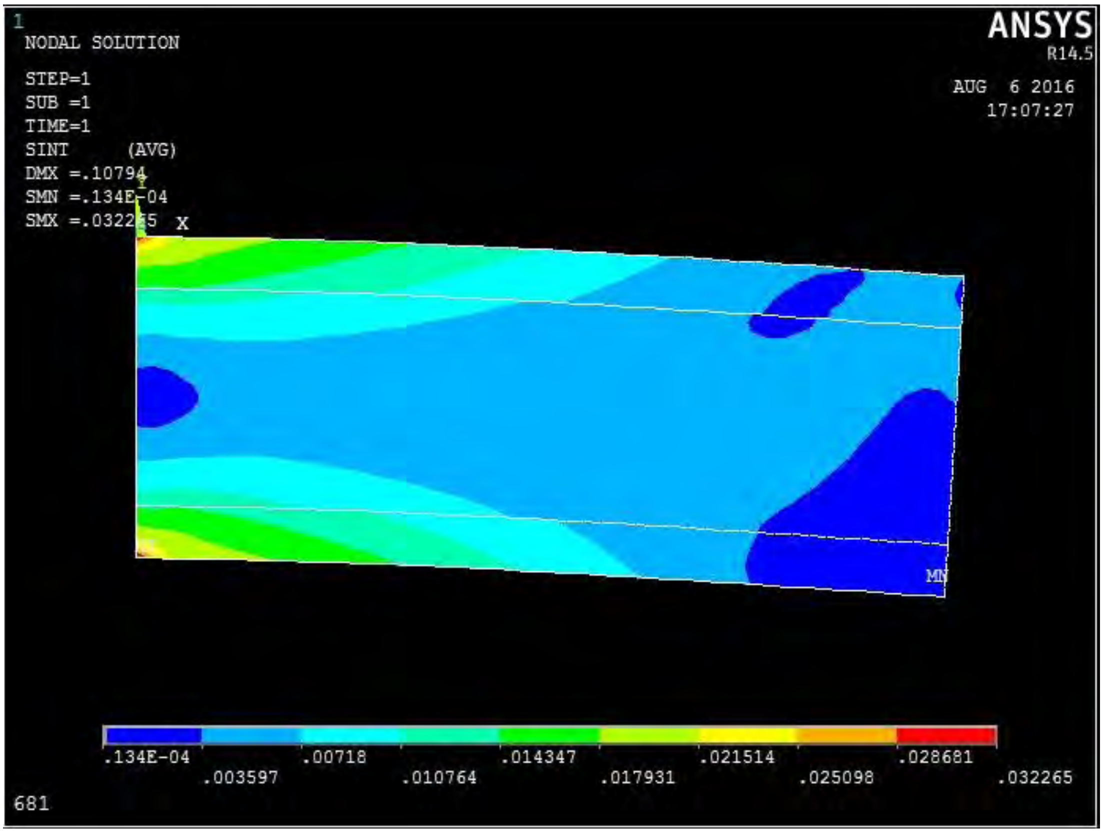


Figure 3.18: Contour Image of deformed FEM model in case of bending

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

In this chapter, initially the numerical solution of sandwich structured composite through finite difference scheme are validated by comparison with finite element solution. For specific problem of sandwich-structured composite having same combination of material properties, same boundary conditions and same loading condition; the problem solved by both finite difference method and finite element method and results are compared. This comparison of results established the reliability and accuracy of the proposed technique. After validation, an isotropic material in each layer of sandwich-structured composite is considered under bending (cantilever). After that, mechanical properties of the case and core materials are varied and the obtained results are analyzed. For similar mechanical properties, loading condition also varied and the obtained results are also analyzed. The distribution of stress and displacement for various combinations of material properties having various boundary condition and various loading conditions are obtained. For simplification, all results of stress are normalized by $\sigma_n = 3 \times 10^{-4}$

4.2 Validation of the solutions of Finite Difference Method (FDM) by the solutions obtained by Finite Element Method (FEM)

4.2.1. Validation under uniform axial displacement

A sandwich-structured composite material with uniform axial displacement as shown in Figure 4.1 has been solved for displacement and stress. The problem is considered as plane stress problem. The left side of the sandwich-structured composite material is fixed and the right side is under uniform displacement. The other sides of the material are free surface. So, at the left side $u_n=0$, $u_t=0$, at the right side $u_n= 3.0\text{x}10^{-4}$, $\overline{\sigma}_t = 0.0$; and at the top and bottom surfaces $\overline{\sigma}_n = 0.0$, $\overline{\sigma}_t = 0.0$. E_1, E_2 are the modulus of elasticity, μ_1, μ_2 are the Poissons' ratios of the case and core material respectively. The geometry of the problem is a rectangle having a=9.25 unit where thickness of casing materials are considered as a_1=0.16a and b=23.5 unit. This problem is solved for stress and displacement distribution by using finite difference method and finite element method taking μ_1=0.29, μ_2=0.31. The results obtained are compared to each other.

Number of mesh is selected for both the FDM and FEM analysis are = 47 x 37, mesh dimension h=0.5 and k=0.25.

For the problem Modulus of elasticity of casing materials of the sandwich-structured composite are considered as E_1=1.0 E_r and modulus of elasticity of core material is considered as E_2=0.9E_r where E_r is the ratio of Modulus of elasticity of core and casing materials i.e. E_2/E_1.

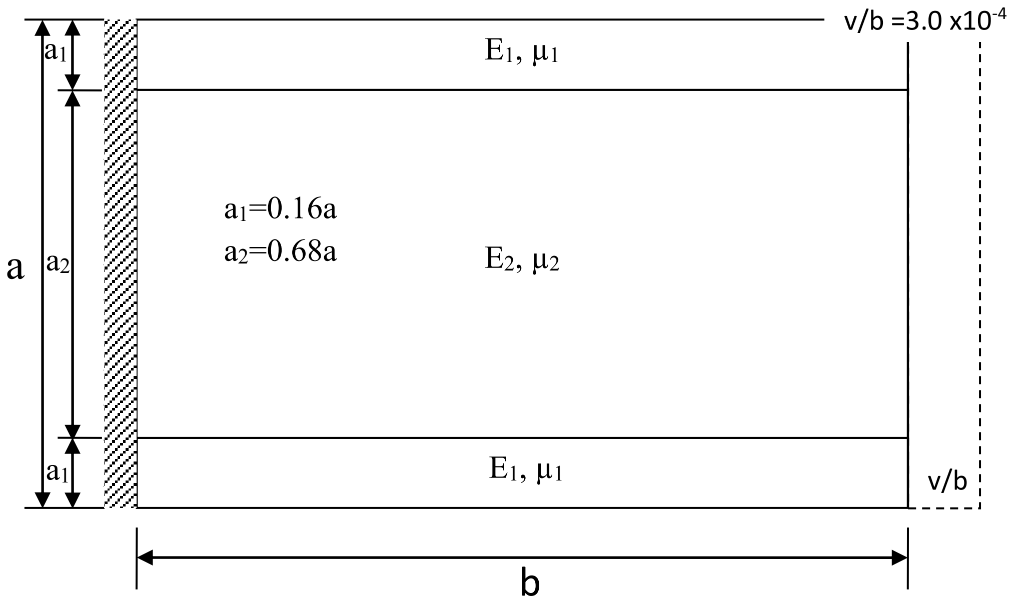


Figure 4.1: Sandwich-structured composite with uniform axial displacement

In both FDM and FEM analysis, u and v are continuous over the layers of a sandwich-structured composite except a slight variation in “v” for right sided region at the interfaces. In case of stress, there are two different values for each stress component, i.e. for upper and lower materials of interfaces of each layer. And there is a discontinuity at the two interfaces of the sandwich-structured composite.

Figure 4.2 shows the comparison of displacement (u/a) distribution at various sections of sandwich-structured composite obtained by both FDM and FEM analysis. At y/b=0.00, two results are zero as the fixed boundary assigned at left side. At the right sided sections from the left, value of the displacement component is increased and the graph is almost similar for FDM and FEM curves. For every section, the curves are symmetrical because material properties of upper and lower material are same as the model is sandwich-structured composite. For the right most section, the slop of the curve is greater.

The distributions of displacement component (v/b) as shown in Figure 4.3 have also similar results for FDM and FEM analysis. There is comparatively more significant variation found at the interfaces by FDM method especially for the sections near right boundary i.e. at y/b=0.75 and y/b=1.00.because of the interface treatments of FDM modeling.

Figure 4.4 shows distribution of normal stress component σ_x/σ_n for different sections of the model by both FDM and FEM method. From the Figure it is depicted that, at the left boundary normal stress σ_x/σ_n is maximum and the results by two methods vary slightly.

The distribution of normal stress σ_y/σ_n at various sections of the sandwich-structured composite is illustrated in Figure 4.5. There is a significant jump found in the Figure at the interfaces of the layers of sandwich-structured composite. Both FDM and FEM method depicts similar results with small differences especially at the fixed boundary regions.

In all Figures, the distributions of the stresses and displacement components are perfectly symmetrical about the center of the sandwich-structured composite model as they should be. The same computer program is used for all the FDM analyses used for different combinations of mechanical properties of the sandwich-structured composite model. This validates that the computer program used for the analysis of sandwich-structured composite is correct.

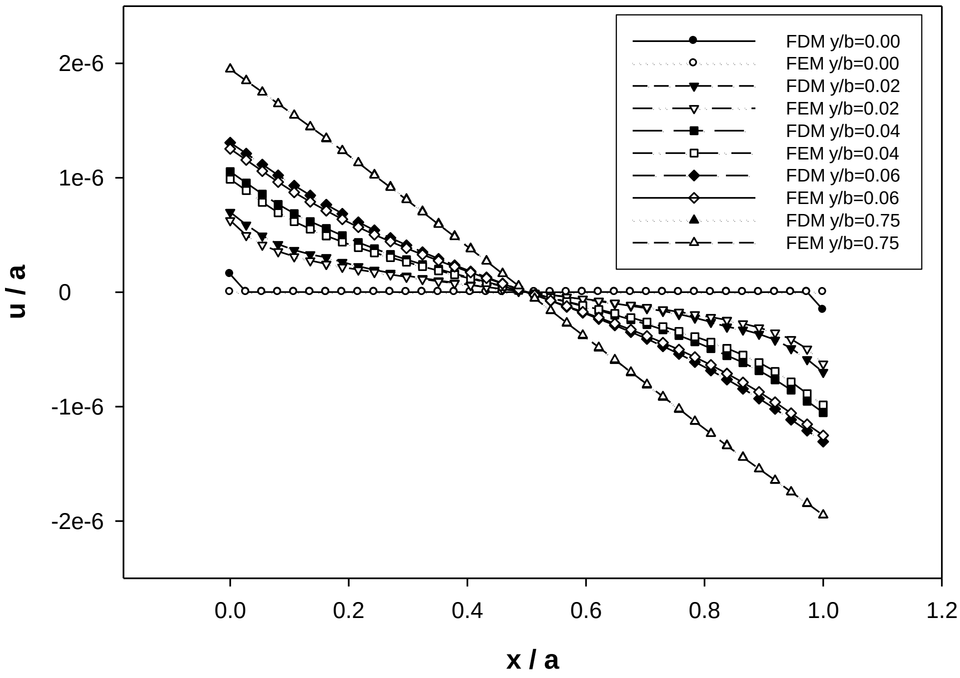


Figure 4.2: Comparison of normalized displacement (u/a) distribution at various sections of the Sandwich-Structured Composite

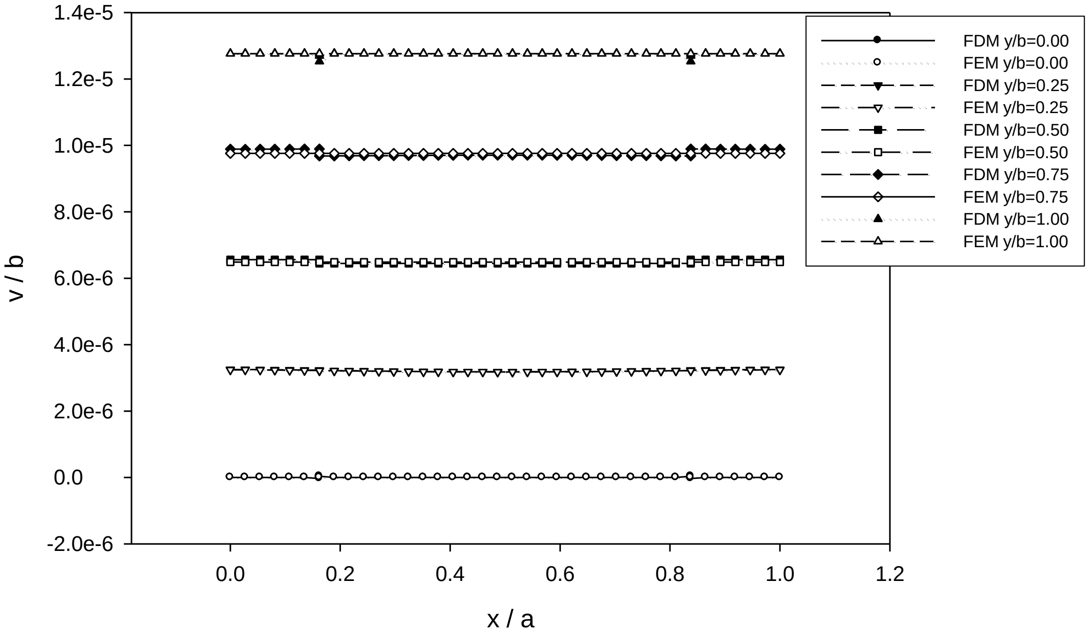


Figure 4.3: Comparison of normalized displacement (v/b) distribution at various sections of the material

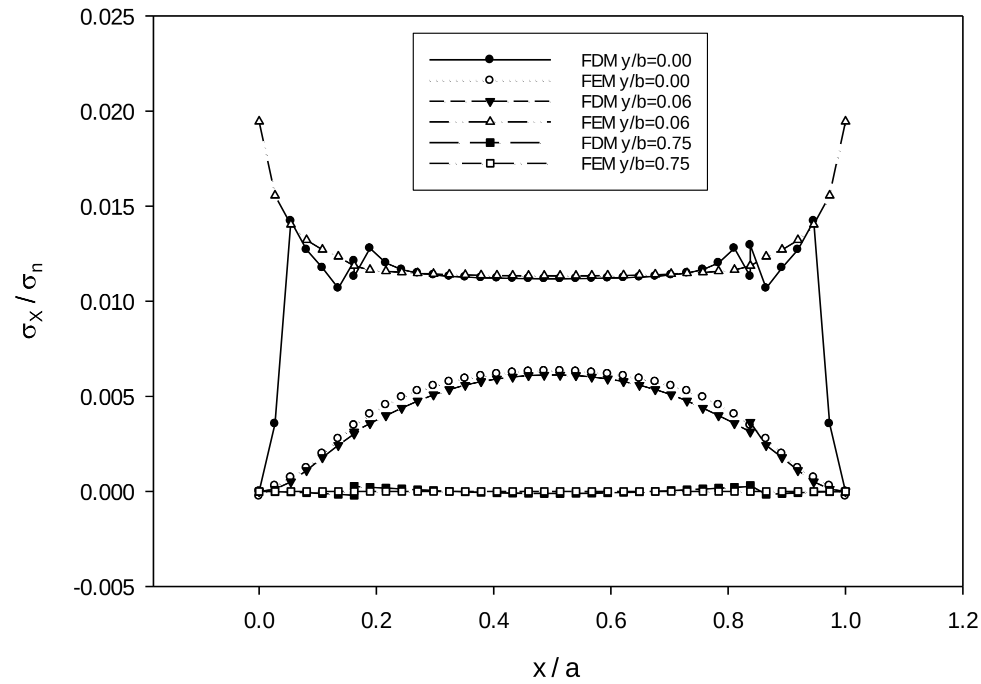


Figure 4.4: Comparison of normalized normal stress (σ_x / σ_n) distribution at different sections of the Sandwich-Structured Composite.

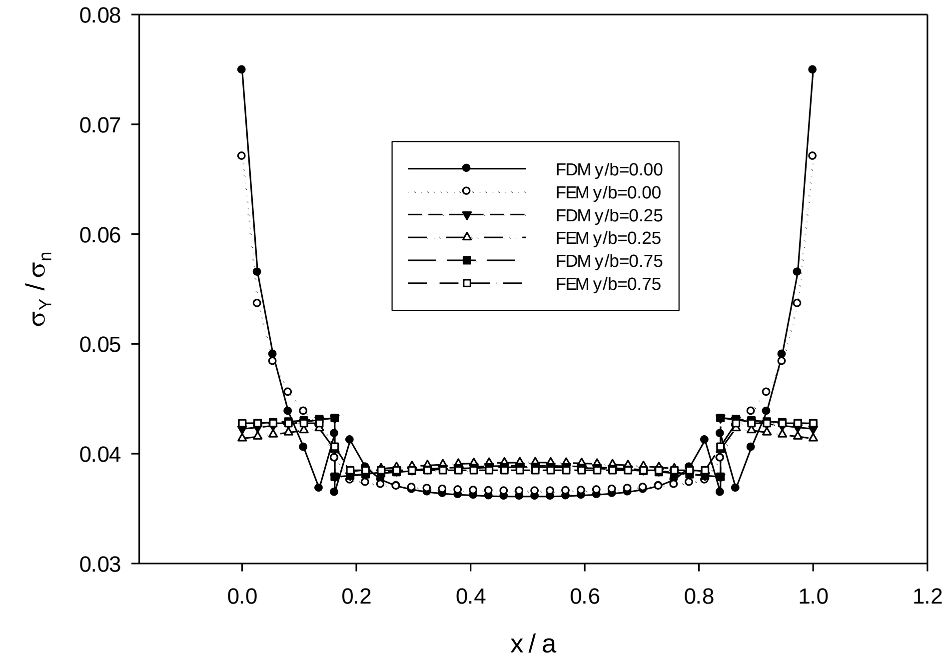


Figure 4.5: Comparison of normalized normal stress (σ_y / σ_n) distribution at different sections of the Sandwich-Structured Composite.

4.2.2. Validation under bending

Here, similar material combination with same dimension is considered under bending. A bending stress σ=3.0x10^{-4} is assigned at five nodes of the right side of top surface. As the sandwich-structured composite object is placed at a fixed end at left and at right it has specific bending load, it may be considered as cantilever beam.

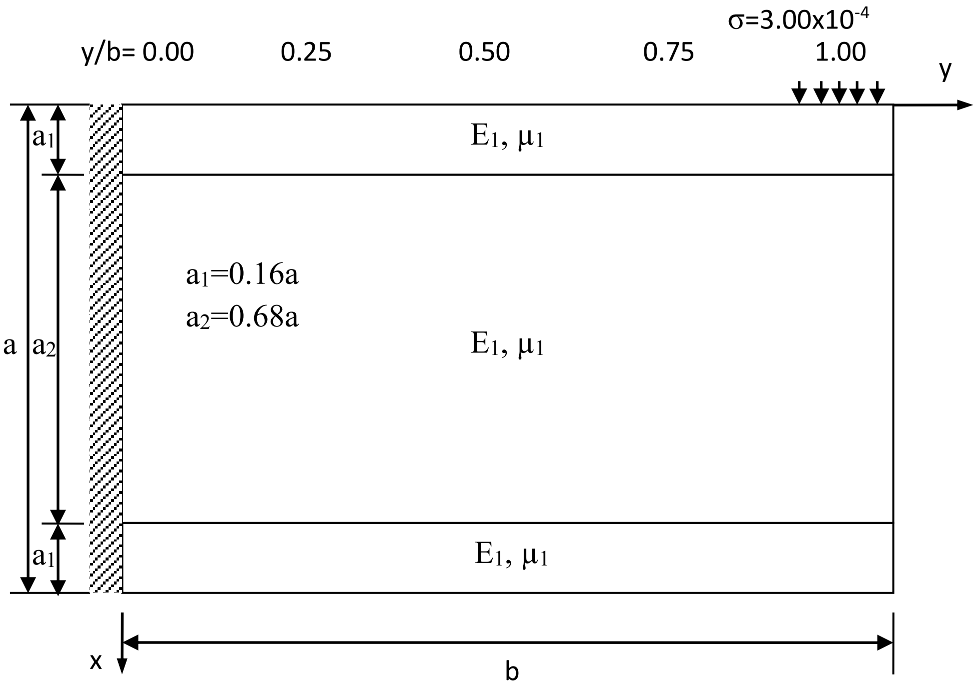


Figure 4.6: Sandwich-structured composite under bending.

The sandwich-structured cantilever model is depicted in Figure 4.6. Figure 4.7 shows the comparison of displacement (u/a) distribution at various sections of sandwich-structured composite obtained by both FDM and FEM analysis. Results obtained by two methods are almost same except the first and last point of each graph because, boundary conditions assigned at the top and bottom left edge are different (displacement parameter is zero in case of FEM modeling but at those specific two points displacement parameter is non-zero in case of FDM modeling).

The comparison of displacement distribution (v/a) at various sections of sandwich structure composite is illustrated in Figure 4.8. The graphs are also similar with a variation at initial and end points and the reason is stated in earlier para.

Figure 4.9 shows distribution of normal stress component σ_x /σ_n for different sections of the model by both FDM and FEM method. From the Figure it is depicted that, at the left boundary normal stress σ_x/σ_n is maximum and the results by two methods vary slightly. At right boundary the normal stress is almost zero and graph for both method are same.

The distribution of normal stress σ_y/σ_n at various sections of the sandwich-structured composite is illustrated in Figure 4.10 There is a significant jump found in the Figure at the interfaces of the layers of sandwich-structured composite. Both FDM and FEM method depicts similar results with small differences especially at the fixed boundary regions.

Comparison of normalized shear stress distribution σ_{xy}/σ_n is also illustrated in Figure 4.11 and the results found are almost same.

In all Figures, the distributions of the stresses and displacement components are symmetrical about the center of the sandwich-structured composite model as they should be. The same computer program is used for all the FDM analyses used for different combinations of mechanical properties of the sandwich-structured composite model. This validates that the computer program used for the analysis of sandwich-structured composite is correct.

From the above discussion, it could be accomplished that the FEM and FDM results for the specified problems in Figure 4.1 and 4.6 are mostly consistent to each other; hence the FDM results are verified. For all the problems of sandwich-structured composite, the same computer program is used for the analyses of displacement component distribution, normal stress distribution, shear stress distribution and other results of interest at the interfacial zones of the layers.

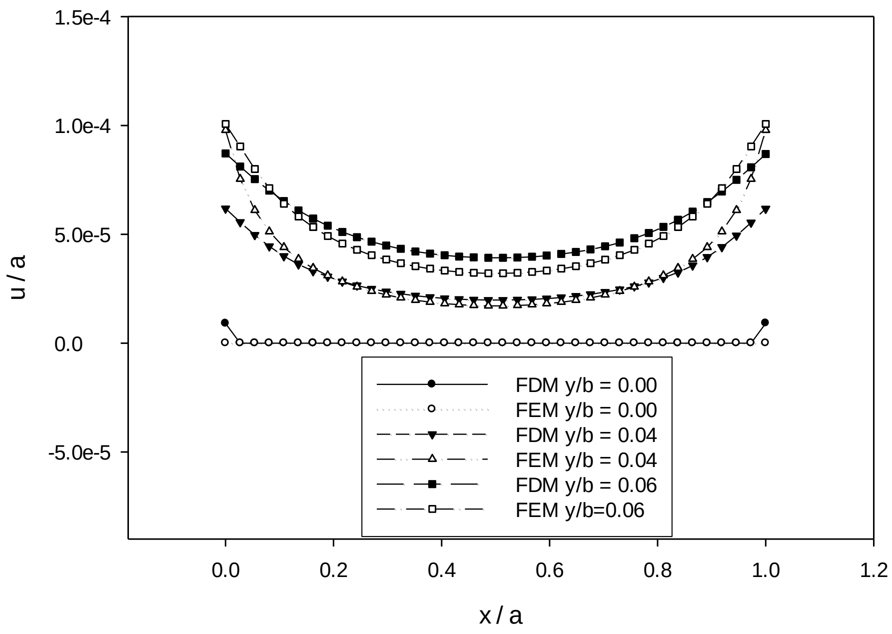


Figure 4.7: Comparison of normalized displacement (u/a) distribution at various sections of the Sandwich-Structured Composite

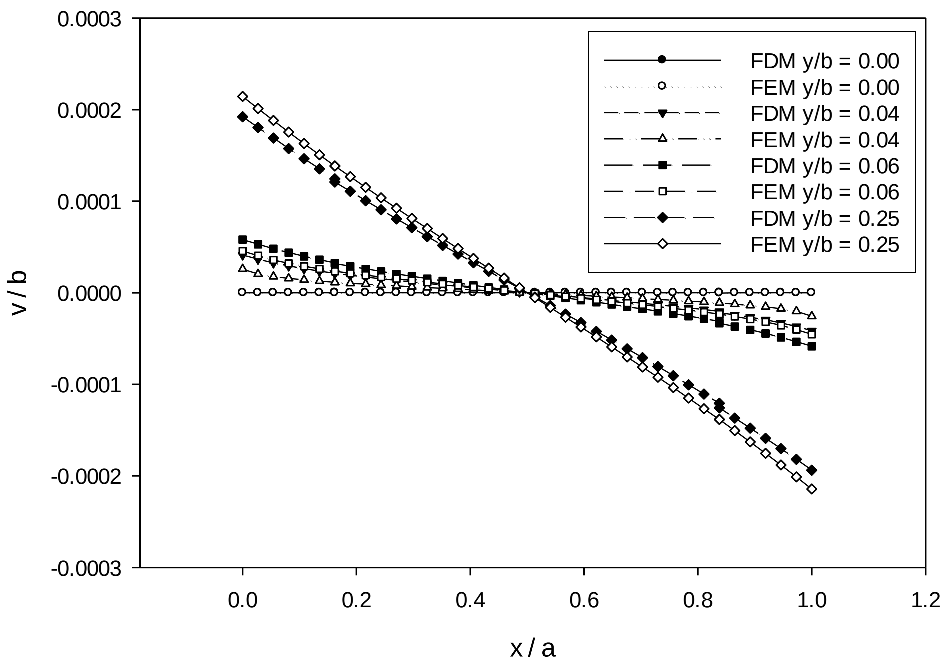


Figure 4.8: Comparison of normalized displacement (v/b) distribution at various sections of the material

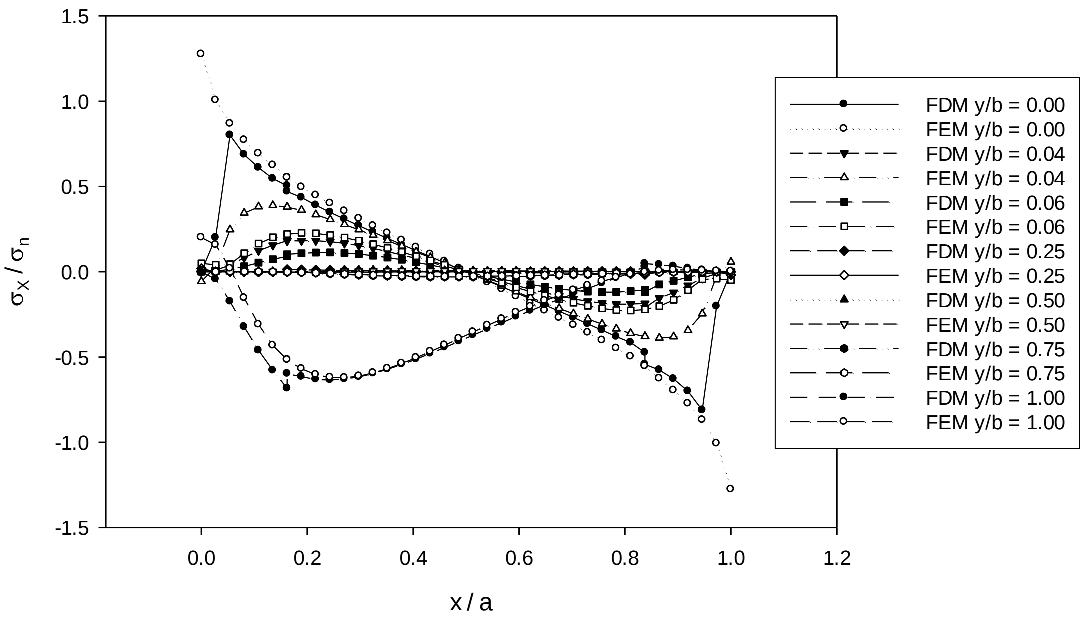


Figure 4.9: Comparison of normalized normal stress (σ_X / σ_n) distribution at different sections of the Sandwich-Structured Composite.

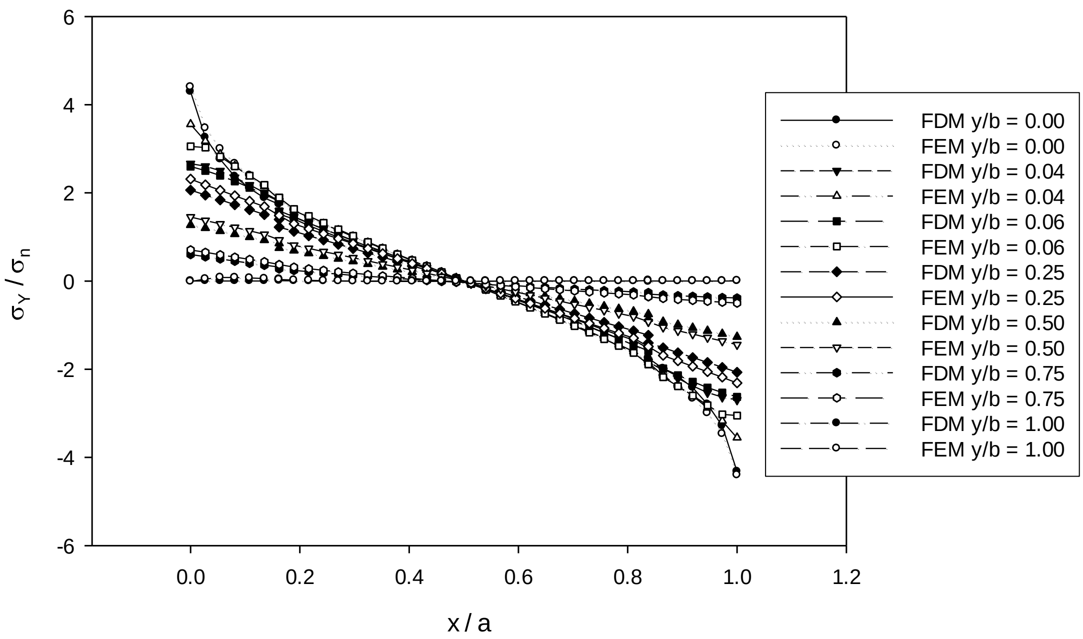


Figure 4.10: Comparison of normalized normal stress (σ_Y / σ_n) distribution at different sections of the Sandwich-Structured Composite.

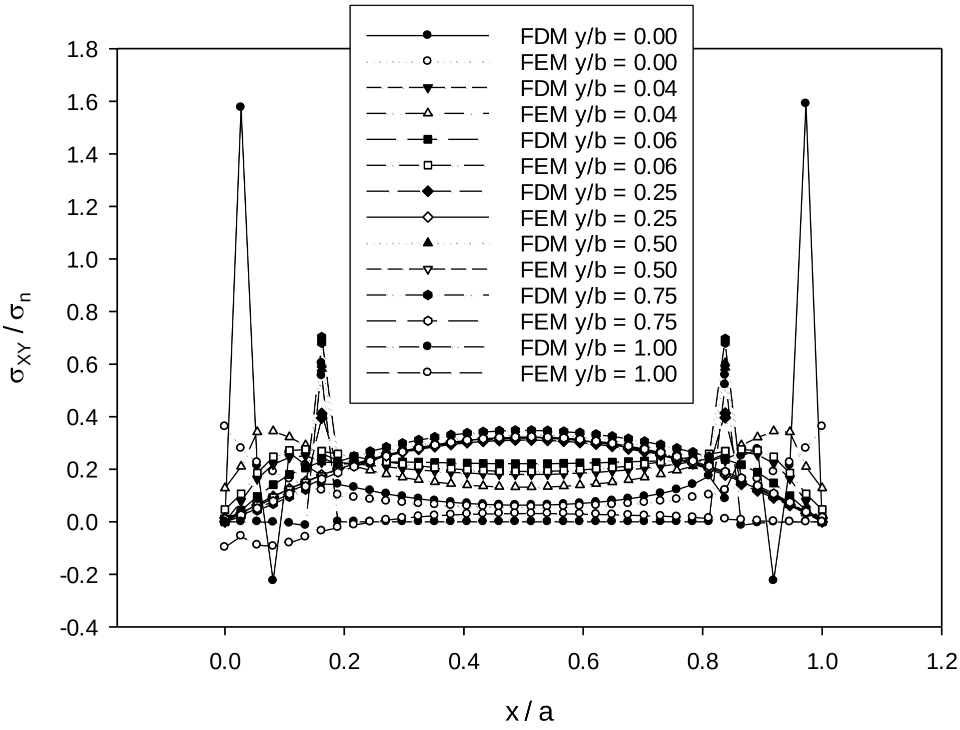


Figure 4.11: Comparison of normalized shear stress (σ_{xy} / σ_n) distribution at different sections of the Sandwich-Structured Composite.

4.3 FDM solution for isotropic material having poisson's ratio μ=0.29 under bending (cantilever).

For steel, Poissons ratio varies from 0.27 to 0.30 and for many other materials, poosson's ratios are almost similar. To analyze the properties of sandwich-structured composite having isotropic layers, firstly an isotropic model with poisons ratio equals to 0.29 throughout the model under bending is considered.

Here the problem is solved taking a=37 and b=47. Modulus of elasticity is also considered same as the material kept under consideration is isotropic in nature. Isotropic solution is considered to compare the solution with sandwich-structured composite.

The two dimensional isotropic model of rectangular geometry is subjected to bending stress at the right side and is fixed at the left side as shown in Figure 4.12. The other two sides are free. Using the proper boundary conditions and formulations as mentioned in chapter 3, the following results are obtained.

The distribution of displacement component ‘u’ is shown in Figure 4.13. The Figure depicts that, value of displacement component ‘u’ is equal to zero at left boundary i.e. y/b=0.00 which conforms the applied fixed boundary condition at left. Graphs for every value of y/b are almost linear but the magnitude of displacement component increases with increasing the value of y/b. The value of displacement component ‘u’ is maximum at right boundary i.e. y/b=1.00 which also conforms the applied boundary load i.e. cantilever.

Figure 4.14 depicts the normalized displacement component (v/a) distribution of the isotropic model. As shown in Figure 4.14, the variation of displacement component (v) is almost linear with x/a. At section, y/b=0, the displacement is nil as it is assigned as the boundary condition. With increasing the value of y/b till 1.00, the magnitude of displacement component is increasing. Though the value of magnitude of displacement varies largely near the left boundary than right boundary, i.e. it can easily segregate the graph for y/b=0.25 and y/b=0.50 but it hardly differs in the graph for y/b=0.75 and y/b=1.00. The value of displacement jumps a bit at the interfaces, i.e. for same value of x/a, there is two different value of ‘v’ at the interfaces. The maximum magnitude of displacement component for each layer occurs at the free surface i.e at x/a=0.00 for upper material and x/a=1.00 for lower material.

The variation of normalized normal stress (σ_x / σ_n) at different sections of a sandwich-structured composite is depicted in Figure 4.15. At the left boundary i.e. y/b=0.00, distribution of normalized normal stress is linear at the center region of the model. At the upper and lower regions of the model, the distributions are non-linear but having large magnitude of stress. Maximum magnitude found near the Top-Left and Bottom-Right boundary. Main differences in the curve of left and right boundary are the magnitude of stresses is positive in upper regions for the left boundary whereas the magnitude is negative in upper regions for the right boundary. At the same time, magnitude of stresses is negative at the lower regions of left boundary whereas the magnitude is positive at the lower regions of right boundary. The graph for values of y/b=0.25, y/b=0.50 and y/b=0.75 could hardly be segregated. The value of stress ‘σ_x’ is almost zero for the maximum portion of these graphs.

Figure 4.16 depicts the normal stress (σ_y/σ_n) distribution at various sections of the isotropic model. The magnitude of stresses is maximum at the upper and lower edges of the body. At the right boundary, the magnitude is also zero which conforms the applied boundary conditions i.e. right side is free. As the graph reaches the left boundary, i.e. y/b reduces from 1.00 to 0.00, the magnitude of stress is increasing. At the left boundary, the magnitudes of stress 'σ_y/σ_n' are maximum. The magnitudes of stress are larger in upper and lower regions than the middle regions of the model. Overall, there is a downward slope for every curve from top to bottom portion of the model.

Normalized shear stress (σ_{xy}/σ_n) distribution at various sections of the material is shown in Figure 4.17. The magnitudes of shear stress are very small at the middle region with a change in the outer regions i.e. upper and lower regions. The graph for left and right boundaries i.e. y/b=0.00 and 1.00 are almost linear for the middle regions with a slight concave shape while the graph for y/b=0.25, y/b=0.50 and y/b=0.75 are convex shaped for core material. At the upper and lower regions, the magnitude varies drastically with abrupt change in the left boundary.

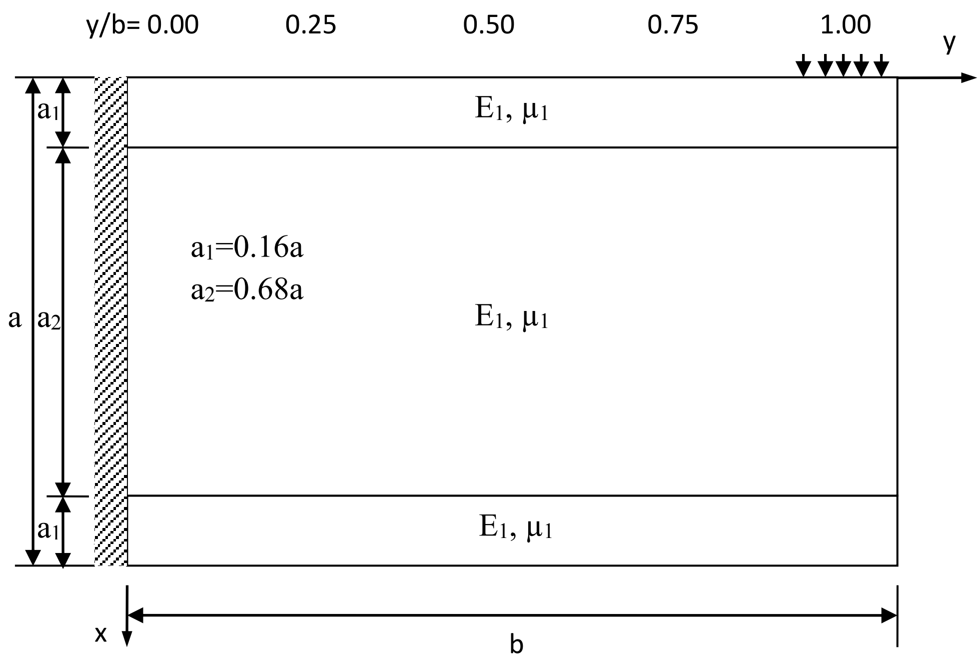


Figure 4.12: Physical elastic problem under bending (cantilever).

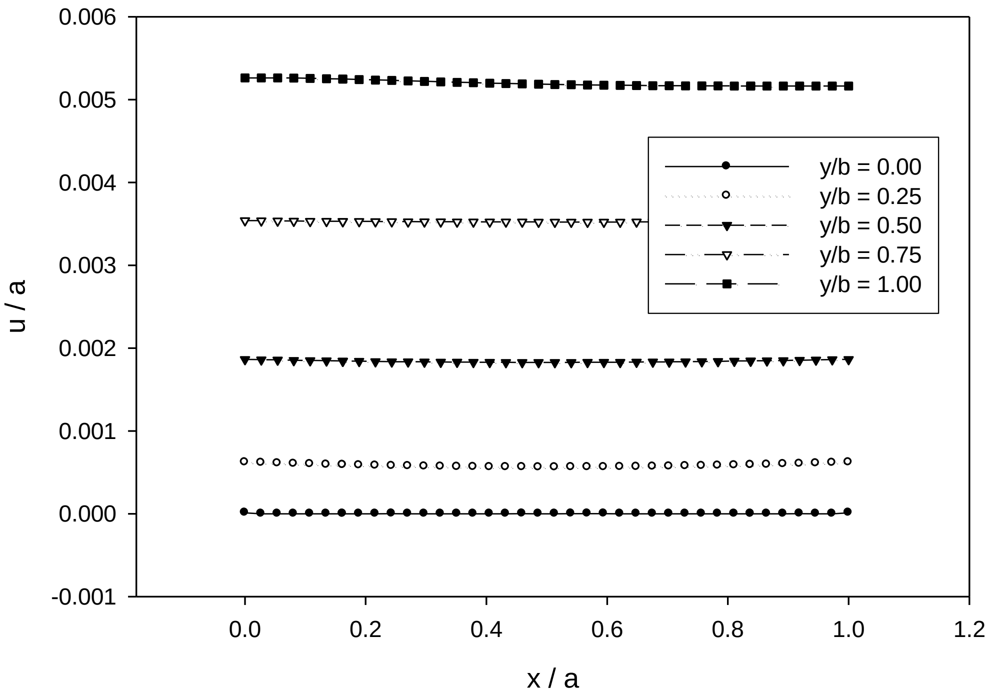


Figure 4.13: Normalized displacement component (u/a) distribution at different sections of an isotropic material.

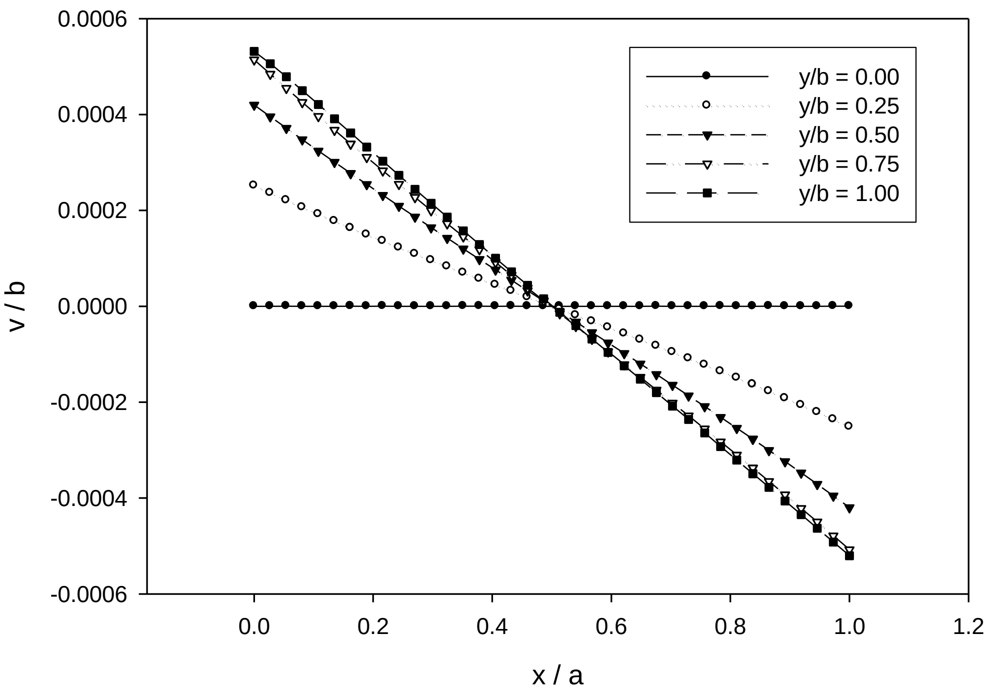


Figure 4.14: Normalized displacement component (v/b) distribution at different sections of an isotropic material.

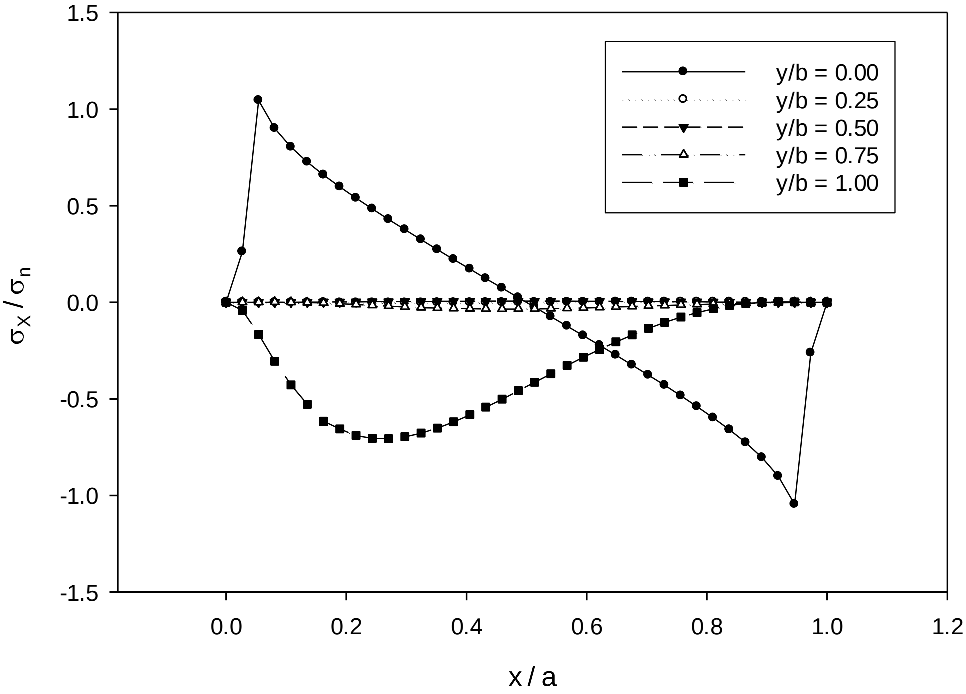


Figure 4.15: Normalized normal stress (σ_X / σ_n) distribution at different sections of an isotropic material.

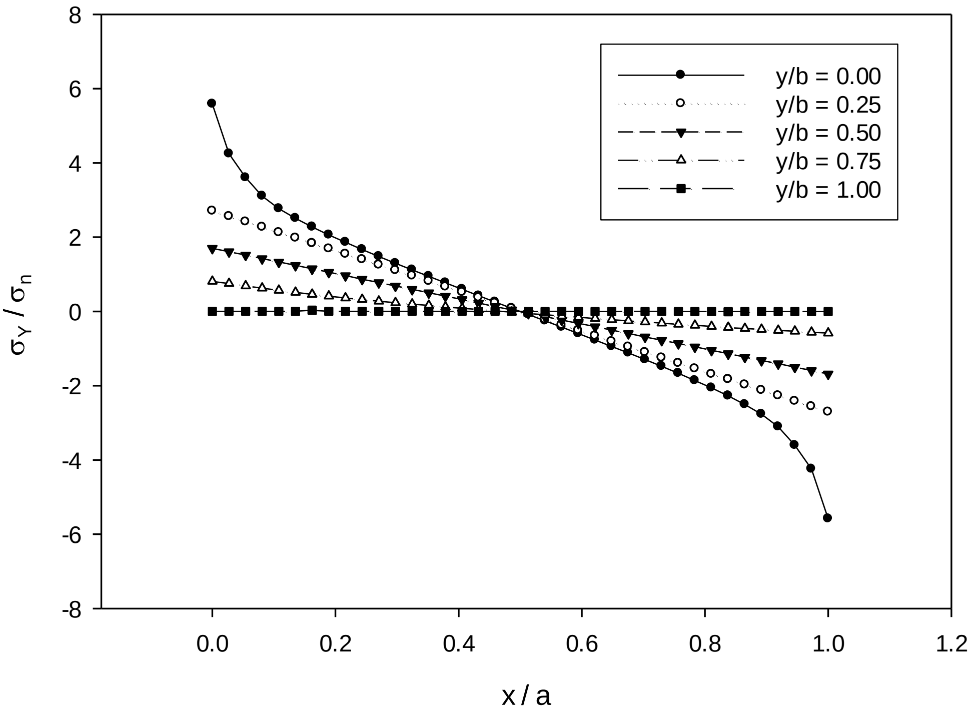


Figure 4.16: Normalized normal stress (σ_Y / σ_n) distribution at different sections of an isotropic material.

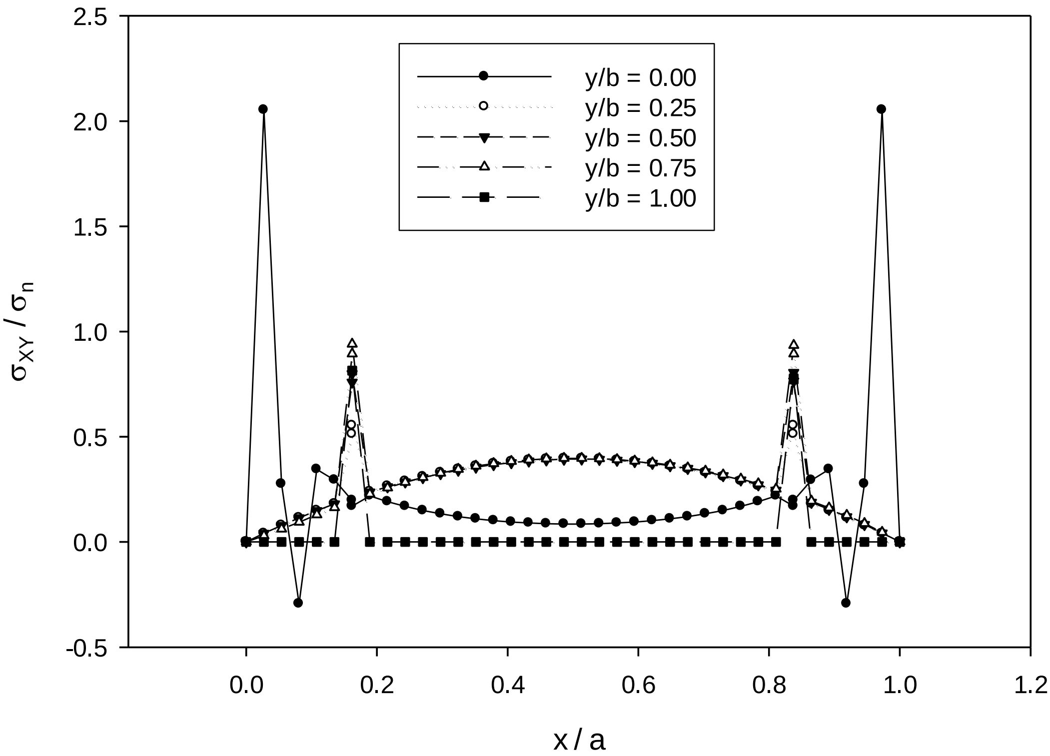


Figure 4.17: Normalized shear stress (σ_{xy} / σ_n) distribution at different sections of a an isotropic material.

4.4 FDM solution for sandwich-structured composite under bending

For pragmatic purpose, detail analyses of displacement and stresses are shown in the next section of the dissertation taking μ_1=0.29 and μ_2=0.31. The problem is solved taking a=37 unit b=47 unit as the dimension of the body. For simplification of analysis, the values of E are normalized and for the particular problem, ratio of the modulus of elasticity E_2/E_1 is taken as 0.90.

The two dimensional sandwich-structured composite of rectangular geometry is subjected to cantilever stress at the right side and is fixed at the left side as shown in Figure 4.6 where other two sides are free. Using the proper boundary conditions and formulations as mentioned in chapter 3, the following results are obtained.

The distribution of displacement component ‘u’ is shown in Figure 4.18. The Figure depicts that, value of displacement component ‘u’ is equal to zero at left boundary i.e. y/b=0.00 which conforms the applied fixed boundary condition at left. Graphs for every value of y/b are almost linear but the magnitude of displacement component increases with increasing the value of y/b. The value of displacement component 'u' is maximum at right boundary i.e. y/b=1.00 which also conforms the applied boundary load i.e. cantilever load at boundary.

Figure 4.19 depicts the normalized displacement component (v/b) distribution of the sandwich-structured composite. As shown in Figure 4.20, the variation of displacement component (v) is almost linear with x/a. At section, y/b=0, the displacement is nil as it is assigned as the boundary condition. With increasing the value of y/b till 1.00, the magnitude of displacement component is increasing. Though the value of magnitude of displacement varies largely near the left boundary than right boundary, i.e. it can easily segregate the graph for y/b=0.25 and y/b=0.50 but it hardly differs in the graph for y/b=0.75 and y/b=1.00. The value of displacement jumps a bit at the interfaces, i.e. for same value of x/a, there is two different value of 'u' at the interfaces. The maximum magnitude of displacement component for each layer occurs at the free surface i.e at x/a=0.00 for upper material and x/a=1.00 for lower material.

The variation of normalized normal stress (σ_x / σ_n) at different sections of a sandwich-structured composite is depicted in Figure 4.21. At the left boundary i.e. y/b=0.00, distribution of normalized normal stress is linear at the core material. At the upper and lower casing material of sandwich-structured composite, the distribution are non-linear but having large magnitude of stress. Maximum magnitude found near the Top- Left and Bottom-Right boundary. There is also a significant variation of the values of stresses at the interfaces of layers of the sandwich-structured composite. At the right boundary, i.e. y/b=1.00, magnitude of stress is maximum at the interfaces. Main differences in the curve of left and right boundary are the magnitude of stresses is positive in upper layer for the left boundary whereas the magnitude is negative in upper layer for the right boundary. At the same time, magnitude of stresses is negative at the lower layer of left boundary whereas the magnitude is positive at the lower layer of right boundary. The graph for values of y/b=0.25, y/b=0.50 and y/b=0.75 could hardly be segregated. The value of stress 'σ_x / σ_n' is almost zero for the maximum portion of these graphs. Though there are always two different values of stress at every graph for the interfaces of layers.

Figure 4.22 depicts the normal stress (σ_y/σ_n) distribution at various sections of the sandwich-structured composite. The magnitude of stresses is maximum at the upper and lower edges of the body. At the right boundary, the magnitude is also zero which conforms the applied boundary conditions i.e. right side is free. Towards left boundary i.e. y/b reduces from 1.00 to 0.00, the magnitude of stress is increasing. At the left boundary, the magnitudes of stress 'σ_y/σ_n' are maximum. There is a clear variation in the magnitude of stresses at the interfaces of layers of sandwich-structured composite. The magnitudes of stress are larger in casing material than the core materials of sandwich-structured composite. Overall, there is a downward slope for every curve from top to bottom portion of the sandwich structured composite. The curve is linear for the core material portion of the sandwich-structured composite. Thus, more stress developed at top and bottom portion of left boundary and are at the interfaces of layers of sandwich-structured composite.

Normalized shear stress (σ_{xy}/σ_n) distribution at various sections of the material is shown in Figure 4.22. The magnitudes of shear stress are very small at the core material with a change in the casing materials. The graph for left and right boundaries i.e. y/b=0.00 and 1.00 are almost linear for the core material with a slight concave shape while the graph for y/b=0.25, y/b=0.50 and y/b=0.75 are convex shaped for core material. At the upper and lower materials, the magnitude varies drastically with abrupt change in the left boundary. At the interfaces the values of shear stress also contains two different magnitudes for two different layers.

.

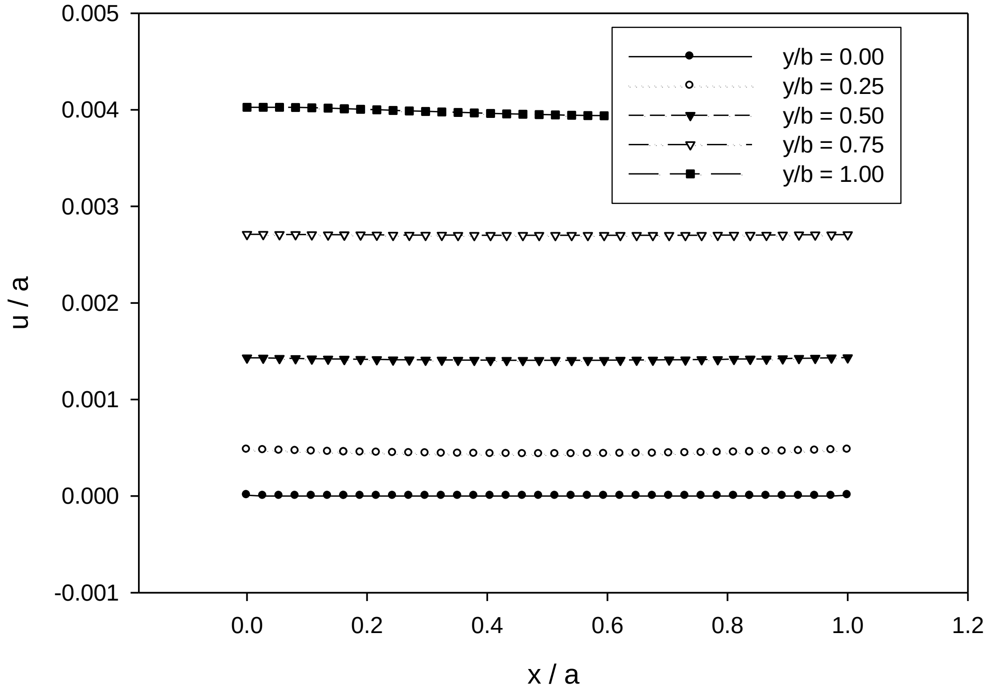


Figure 4.18: Normalized displacement component (u/a) distribution at different sections of a sandwich-structured composite

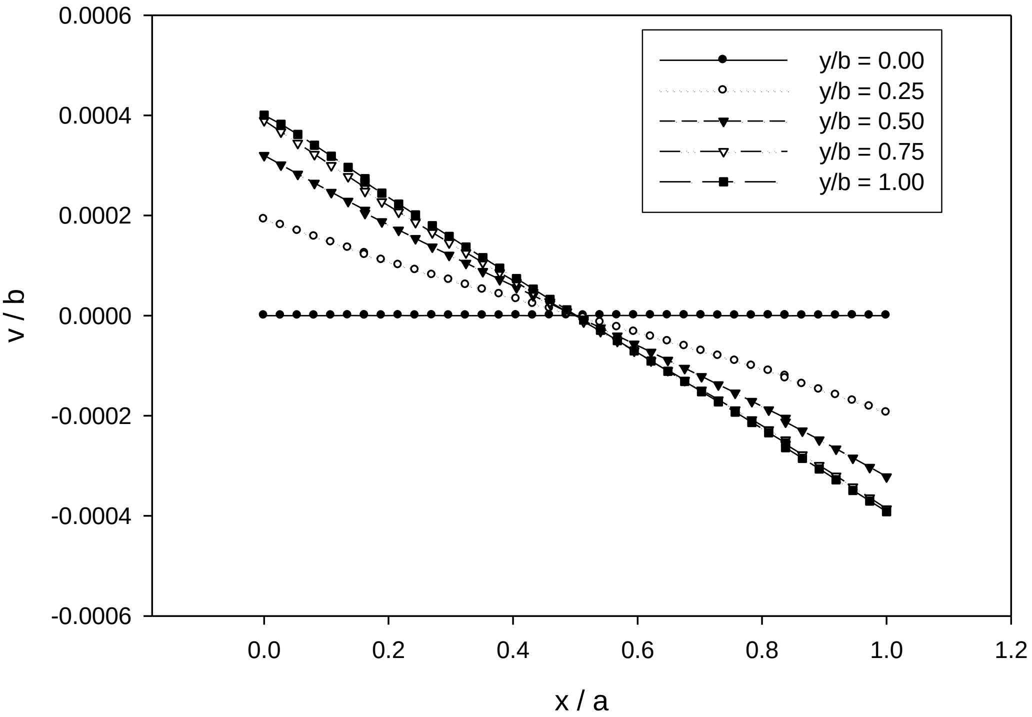


Figure 4.19: Normalized displacement component (v/b) distribution at different sections of a sandwich-structured composite.

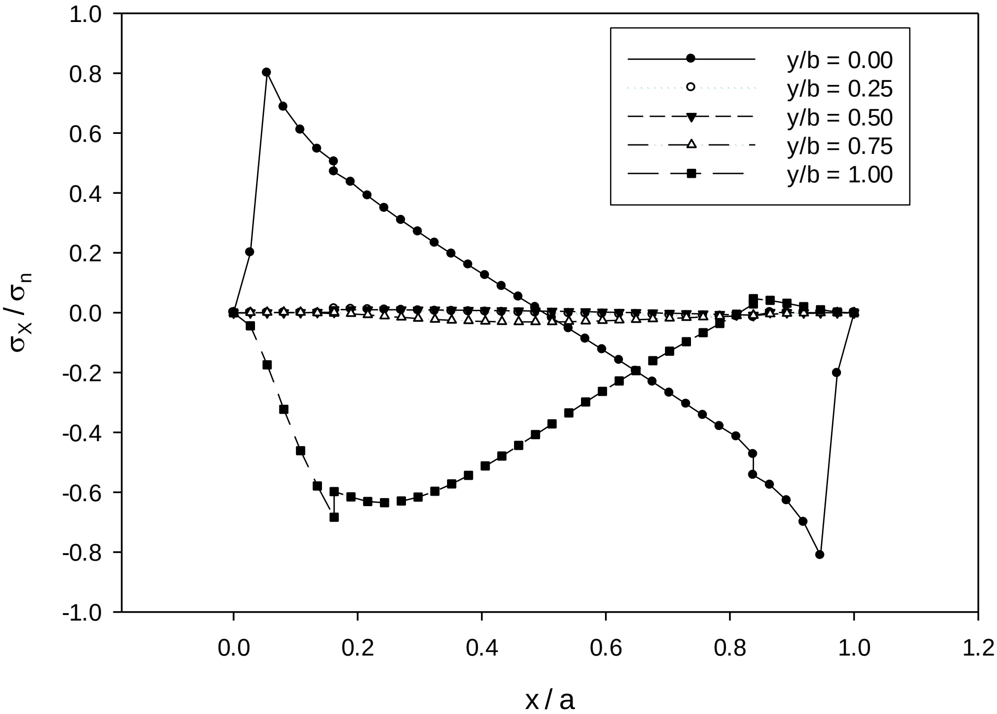


Figure 4.20: Normalized normal stress (σ_x / σ_n) distribution at different sections of a sandwich-structured composite

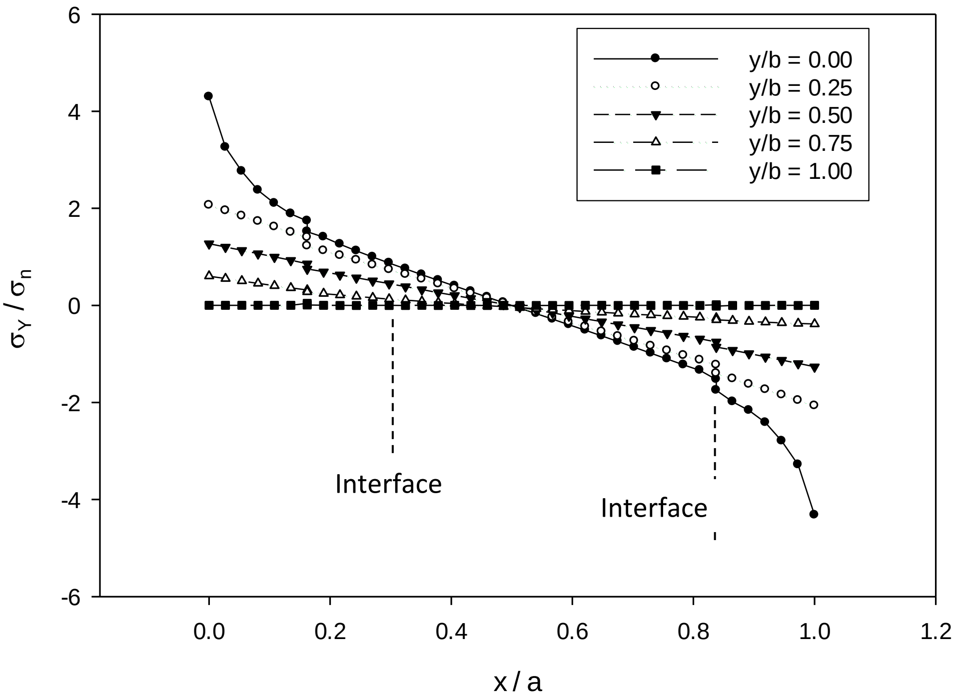


Figure 4.21: Normalized normal stress (σ_y / σ_n) distribution at different sections of a sandwich-structured composite.

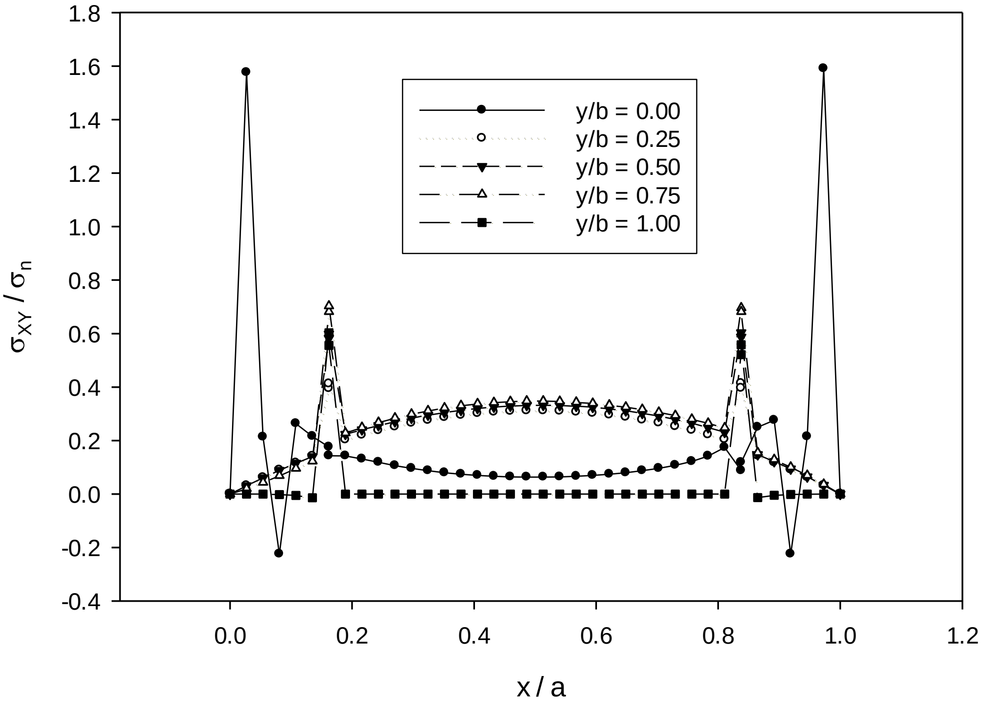


Figure 4.22: Normalized shear stress (σ_{xy}/ σ_n) distribution at different sections of a sandwich-structured composite.

4.5 FDM solution for sandwich-structured composite under uniform axial tension.

To consider the scenario for different loading conditions with the effect of modulus of elasticity, the model of sandwich-structured composite is kept under consideration with uniform axial tension. The combination of material is considered as same poissons ratio throughout the model i.e. $\mu_1 = \mu_2 = 0.29$ but different modulus of elasticity for the layers. The ratio of modulus of elasticity of core material and casing layers are considered as 0.80.Figure 4.23 shows the model.

The distribution of normalized displacement component 'u/a' is illustrated in Figure 4.24. In the Figure it is depicted that, the pattern of each graph is linear. And the magnitudes of displacement component are maximum at the upper and lower regions of the model.

In Figure 4.25, distribution of normalized displacement component 'v/b' is illustrated. Graphs for all sections are almost linear and horizontal with the x-axis. At the right side of the model, graph is slightly convex shaped. That means, for uniform axial tension throughout the body, soft core material elongates more than the comparatively hard case materials.

The variation of normalized normal stress (σ_x/ σ_n) at different sections of a sandwich-structured composite is depicted in Figure 4.26. At the left boundary i.e. y/b=0.00, distribution of normalized normal stress is linear at the core material and found almost horizontal to x-axis. At the upper and lower casing material of sandwich-structured composite, the distribution are non-linear but having large magnitude of stress. There is also a significant variation of the values of stresses at the interfaces of layers of the sandwich-structured composite. At, y/b=0.25, y/b=0.50 and y/b=0.75 the graph depicts normalized normal stress as zero. Though there are always two different values of stress at every graph for the interfaces of layers.

Figure 4.27 depicts the normal stress (σ_y/ σ_n) distribution at various sections of the sandwich-structured composite. The magnitude of stresses is maximum at the upper and lower edges of the body. At the right boundary, the magnitude is equal to the assigned value which conforms the applied boundary conditions i.e. right side the axial tension acts. There is a clear variation in the magnitude of stresses at the interfaces of layers of sandwich-structured composite. The magnitudes of stress are larger in casing material than the core materials of sandwich-structured composite. There is critical zones are at the top and bottom portion of left boundary and are at the interfaces of layers of sandwich-structured composite.

Normalized shear stress (σ_{xy}/ σ_n) distribution at various sections of the material is shown in Figure 4.28. The magnitudes of shear stress are very small at the core material with a change in the casing materials. Shear stress is almost zero throughout the model with some significant change for the top left and bottom left boundaries. At the interfaces the values of shear stress also contains two different magnitudes for two different layers.

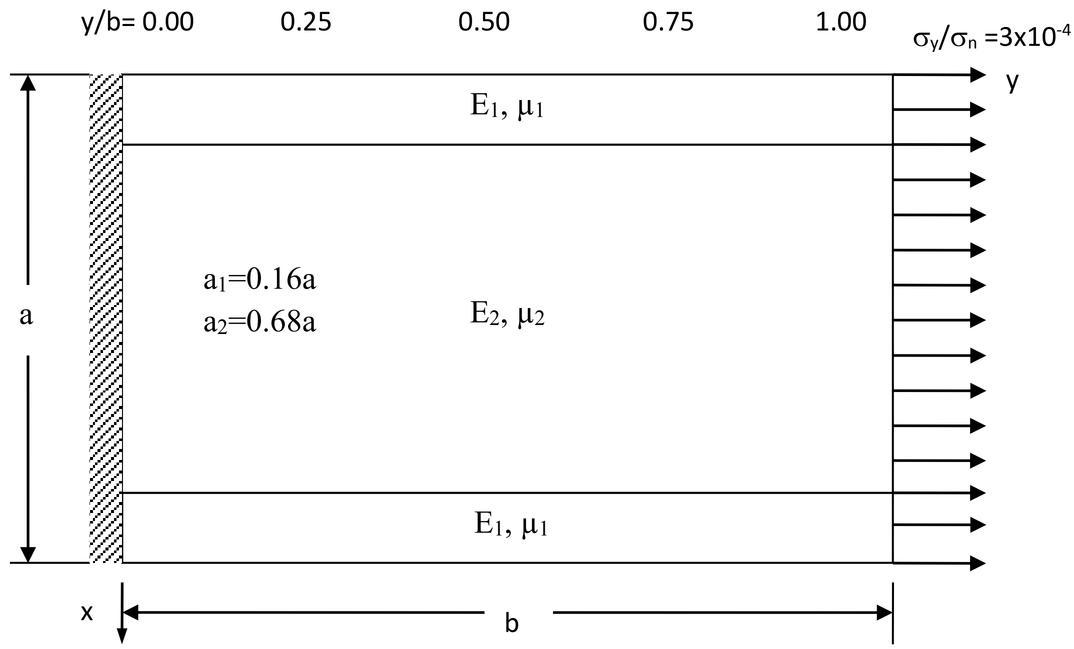


Figure 4.23: Physical elastic problem under axial tension

.

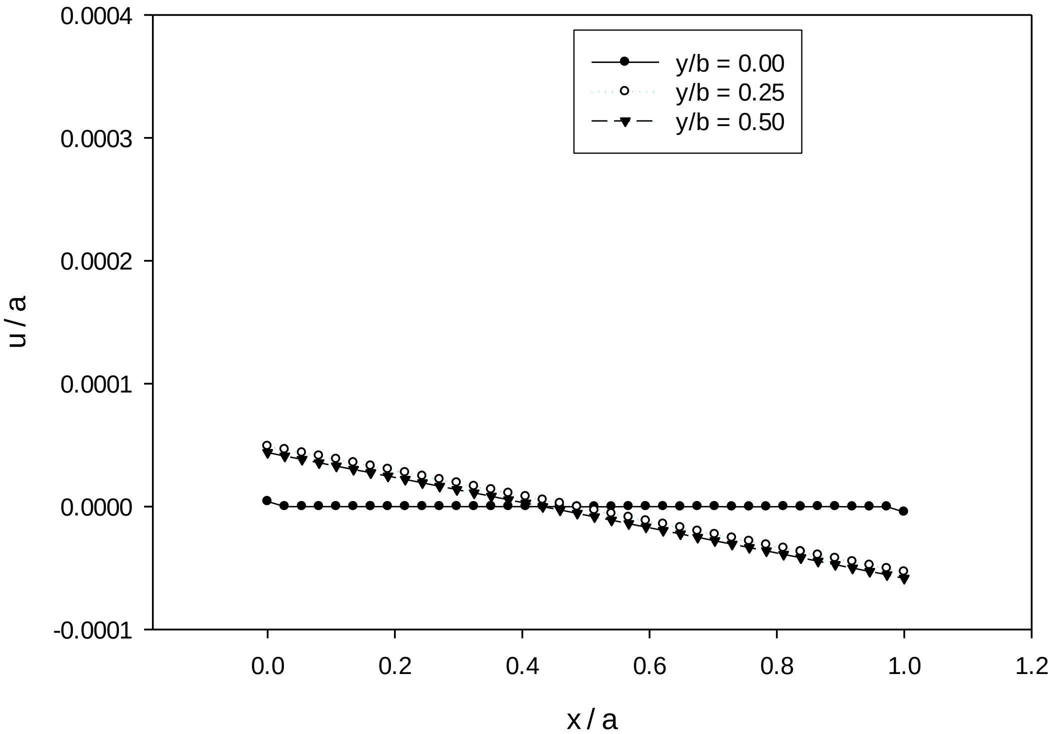


Figure 4.24: Normalized displacement component (u/a) distribution at different sections of a sandwich-structured composite

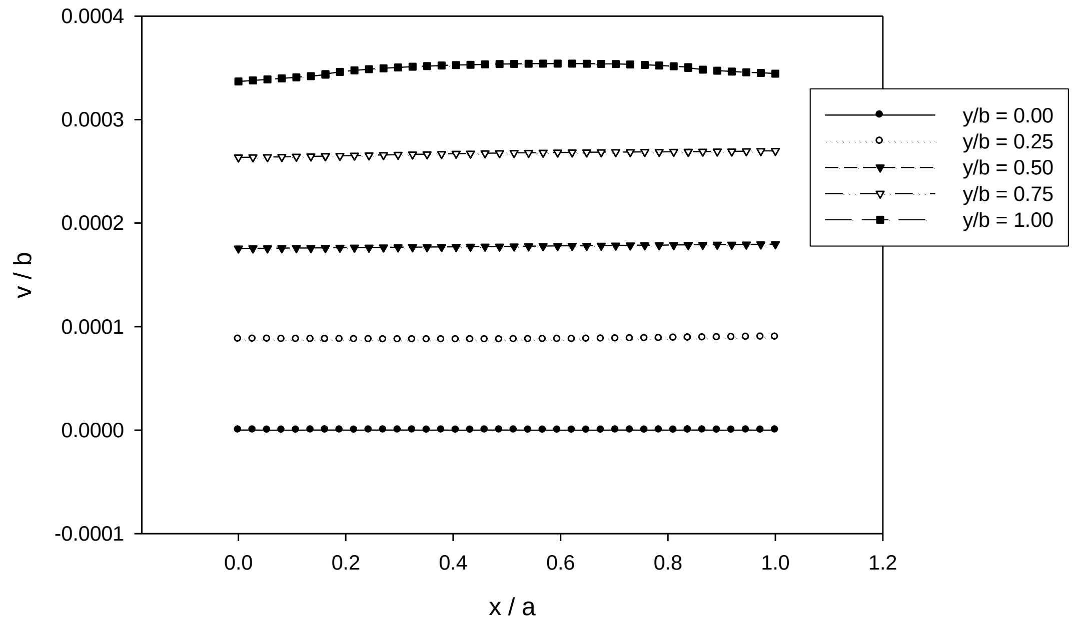


Figure 4.25: Normalized displacement component (v/b) distribution at different sections of a sandwich-structured composite.

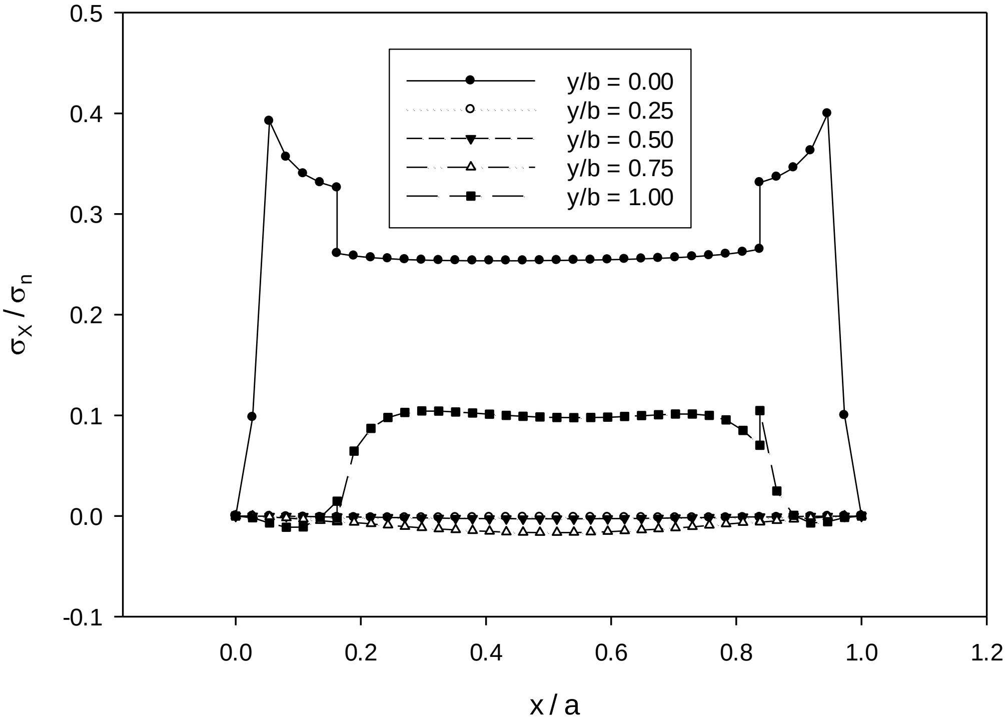


Figure 4.26: Normalized normal stress (σ_x/ σ_n) distribution at different sections of a sandwich-structured composite

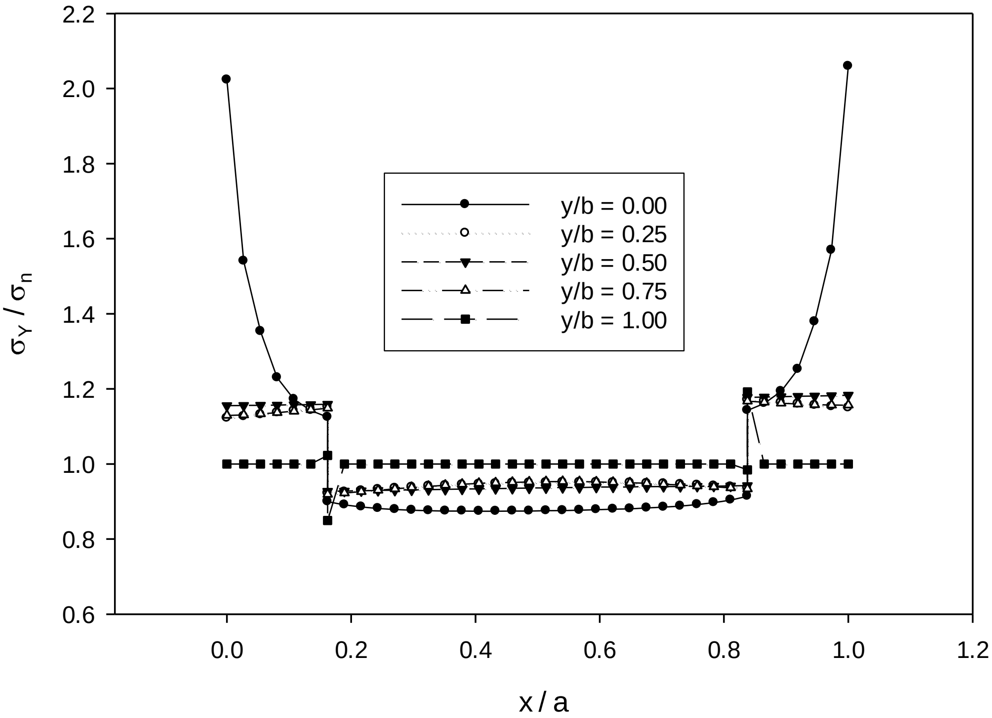


Figure 4.27: Normalized normal stress (σ_y / σ_n) distribution at different sections of a sandwich-structured composite.

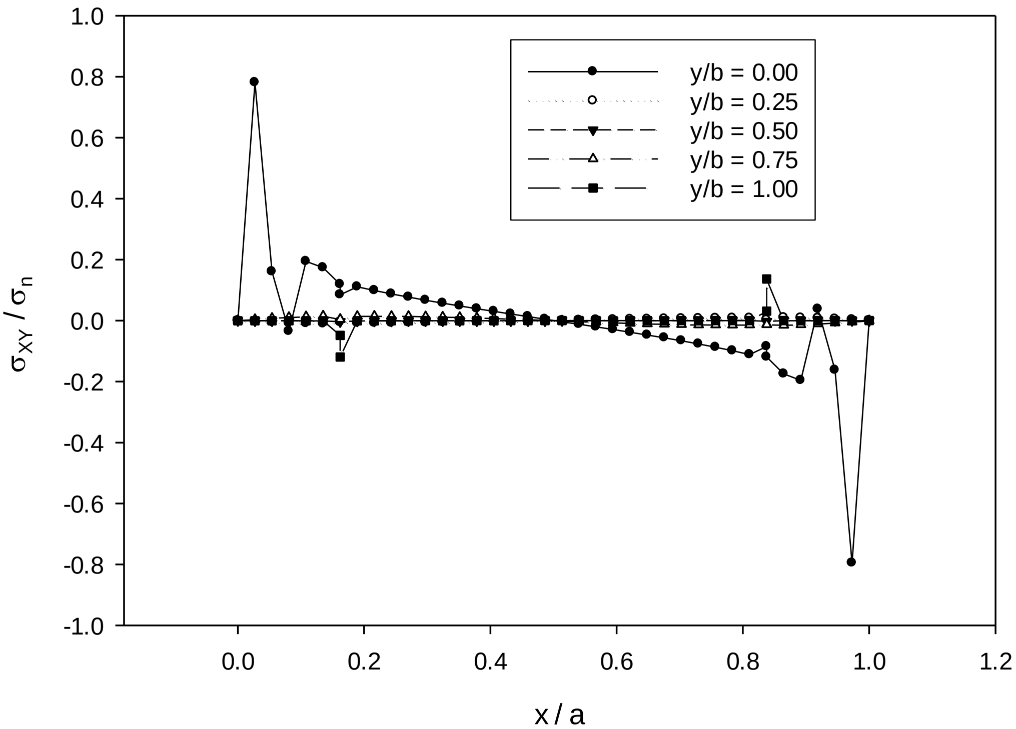


Figure 4.28: Normalized shear stress (σ_{xy}/ σ_n) distribution at different sections of a sandwich-structured composite.

4.6 FDM solution for sandwich-structured composite having uniform axial displacement .

Figure 4.29 shows the comparison of displacement (u/a) distribution at various sections of sandwich-structured composite. At y/b=0.00, results are zero as the fixed boundary assigned at left side. At the right sided sections from the left, value of the displacement component is increased. For every section, the curves are symmetrical because material properties of upper and lower material are same as the model is sandwich-structured composite. For the right most section, the slop of the curve is greater.

The distributions of displacement component (v/b) as shown in Figure 4.30 have also depicted linear curves as shown in curve for axial tension model. All the graphs are almost parallel to horizontal axis. Here variation of magnitude found at the interfaces especially for the sections near right boundary i.e. at y/b=0.75 and y/b=1.00.

Figure 4.31 shows distribution of normal stress component σ_x/σ_n for different sections of the model. From the Figure it is depicted that, at the left boundary normal stress σ_x/σ_n is maximum. At the right boundary the normal stress is almost zero.

The distribution of normal stress σ_y/σ_n at various sections of the sandwich-structured composite is illustrated in Figure 4.32. There is a significant jump found in the Figure at the interfaces of the layers of sandwich-structured composite.

Normalized shear stress (σ_{xy}/σ_n) distribution at various sections of the material is shown in Figure 4.33. The magnitudes of shear stresses are almost zero throughout the layers. At the interfaces the values of shear stress also contains two different magnitudes for different layers of sandwich-structured composite.

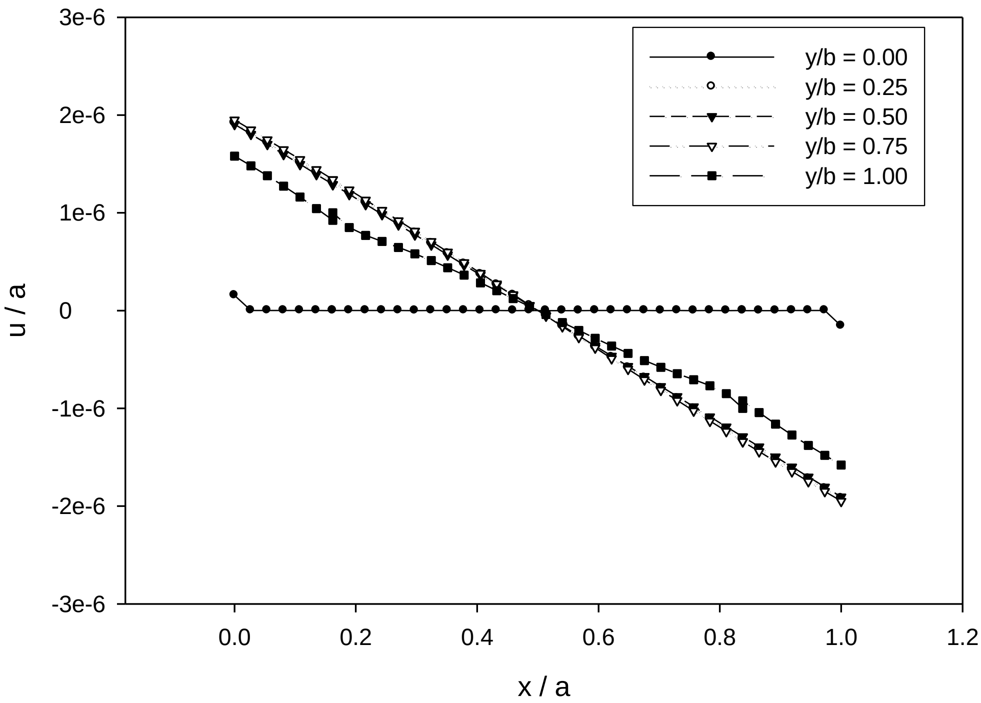


Figure 4.29: Normalized displacement component (u/a) distribution at different sections of a sandwich-structured composite

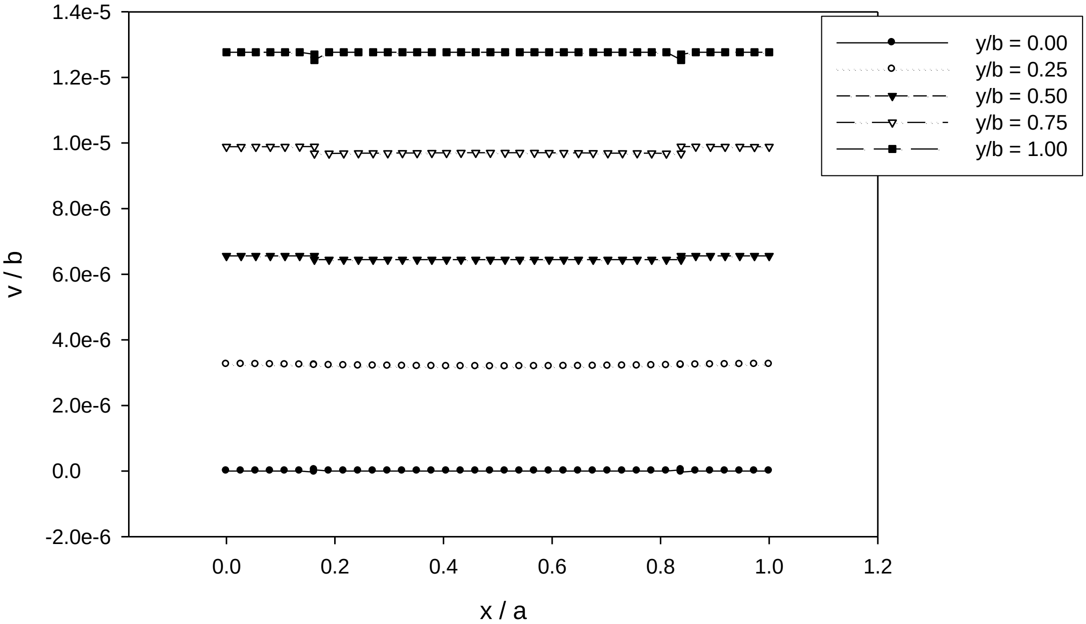


Figure 4.30: Normalized displacement component (v/b) distribution at different sections of a sandwich-structured composite.

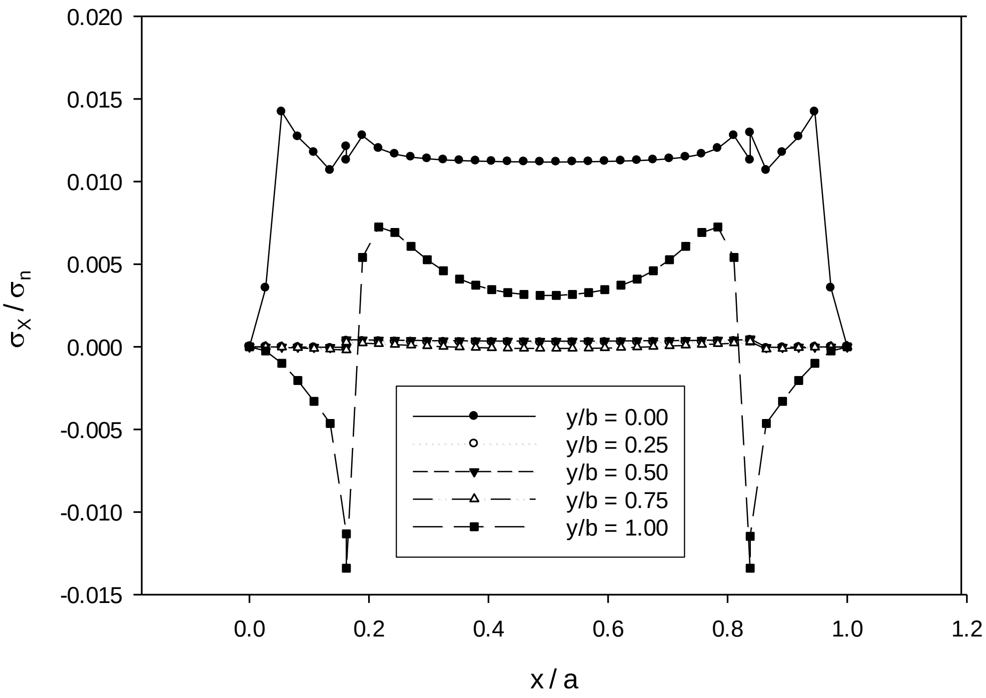


Figure 4.31: Normalized normal stress (σ_X / σ_n) distribution at different sections of a sandwich-structured composite

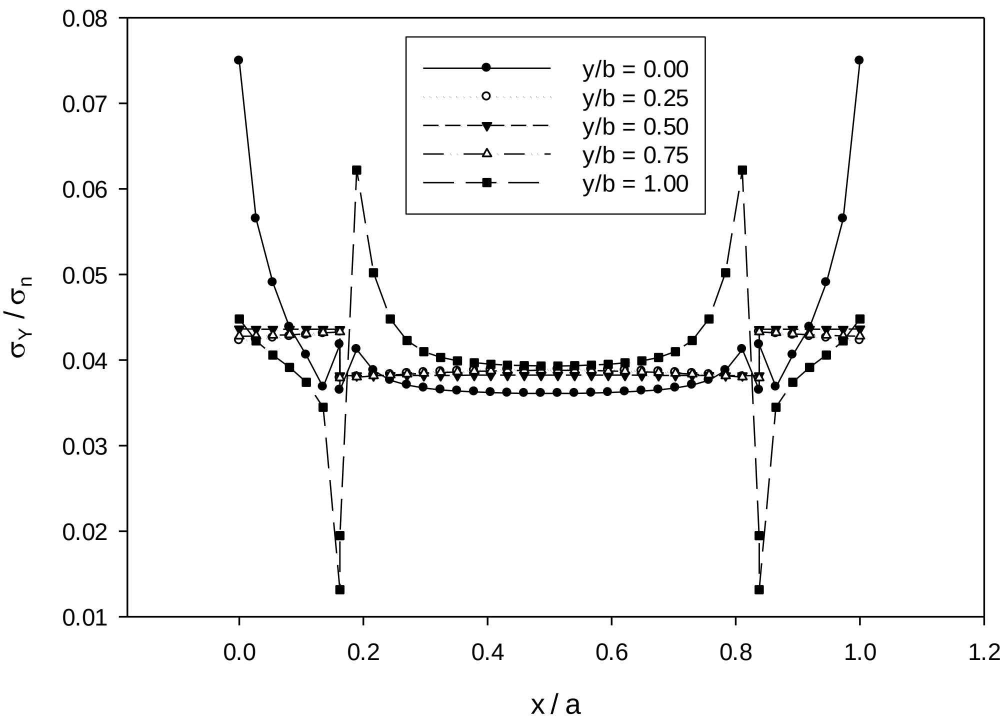


Figure 4.32: Normalized normal stress (σ_y / σ_n) distribution at different sections of a sandwich-structured composite.

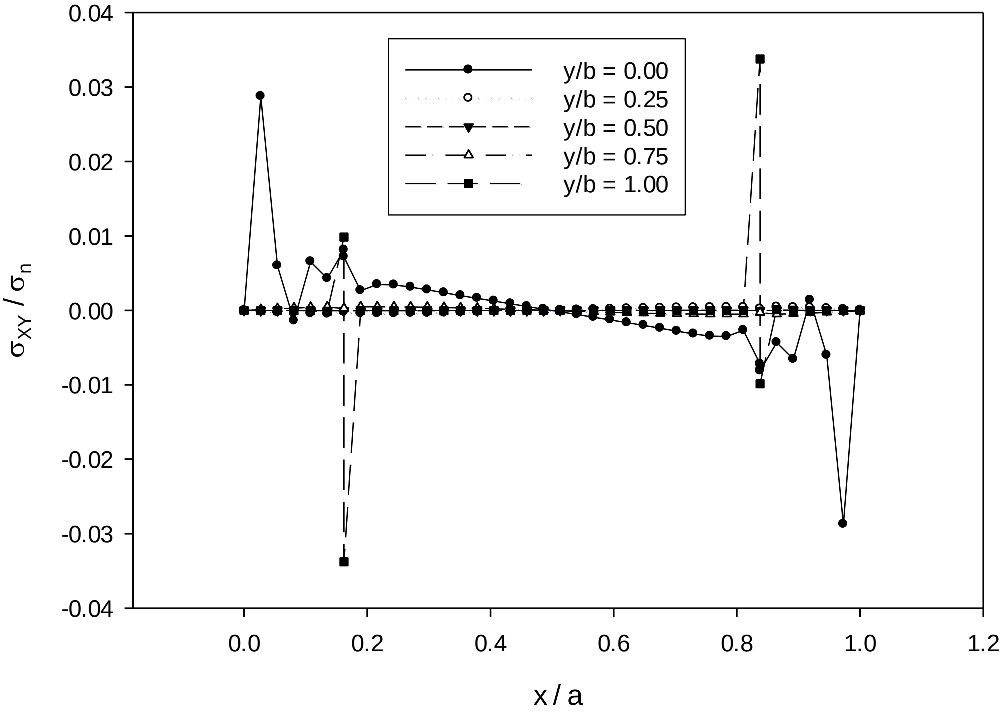


Figure 4.33: Normalized shear stress (σ_{xy}/σ_{n}) distribution at different sections of a sandwich-structured composite.

4.7 Effect of Modulus of Elasticity on FDM solution for sandwich-structured composite having uniform axial displacement.

To analyze the effect of modulus of elasticity, same model of uniform displacement at right boundary are kept into consideration with different modulus of elasticity. As the ratio of modulus of elasticity is taken into consideration for simplification of procedure for solution, 03 different modulus of elasticity ratio are considered. Figure 4.1 shows the model on which the effect of modulus of elasticity is illustrated.

The variation of stress components are found for changing modulus of elasticity. For increasing the modulus of elasticity of core materials, the normalized stress σ_x / σ_n distribution curve became more concave for the core material regions as depicted in Figure 4.34.

In Figure 4.35, it is found that the normalized stress σ_y/σ_n distribution have significant changes due to change in the value of modulus of elasticity for the core materials. As the value of modulus of elasticity reduces, the magnitude of normalized normal stress σ_y/σ_n also reduces.

The comparison of normalized shear stress distribution is illustrated in Figure 4.36. It is found that, as the value of modulus of elasticity increases the magnitude of shear stress also increases which was almost zero for the less modulus of elasticity.

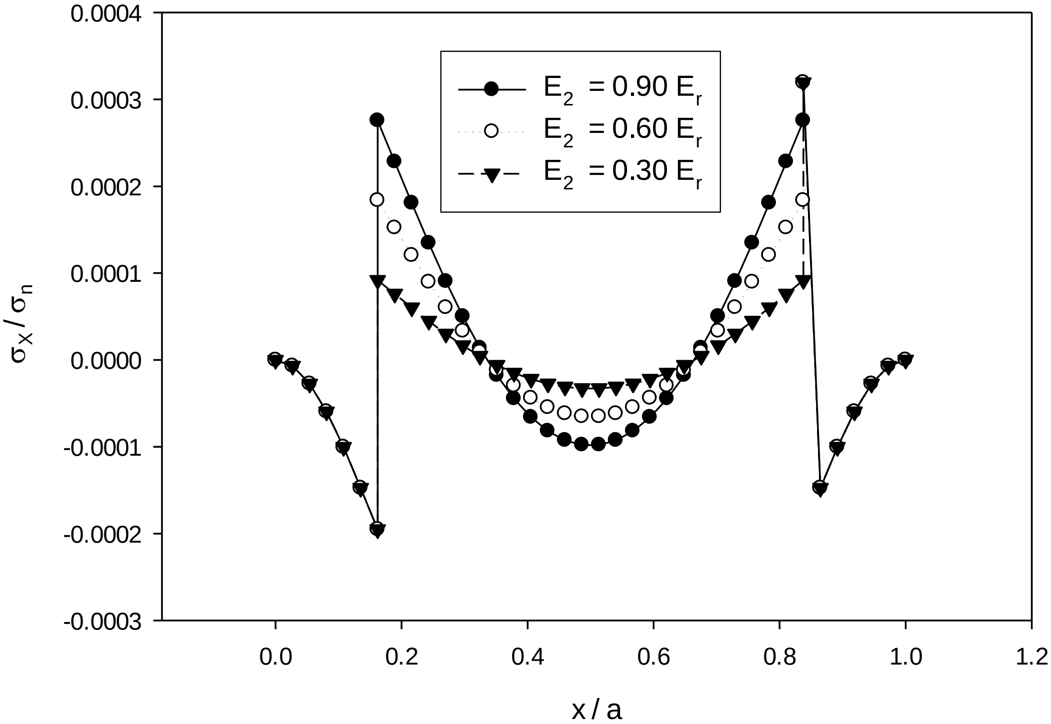


Figure 4.34: Comparison of normalized normal stress ($\sigma_x/\ \sigma_n$) distribution at section y/b=0.75 of a sandwich-structured composite.

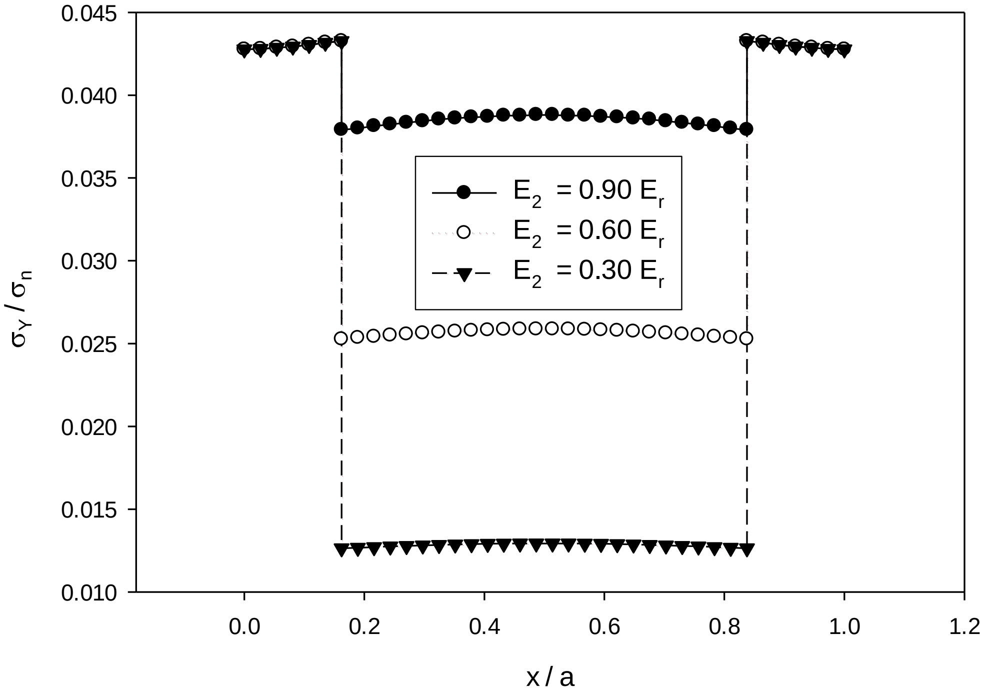


Figure 4.35: Comparison normalized normal stress (σ_y / σ_n) distribution at different sections of a sandwich-structured composite.

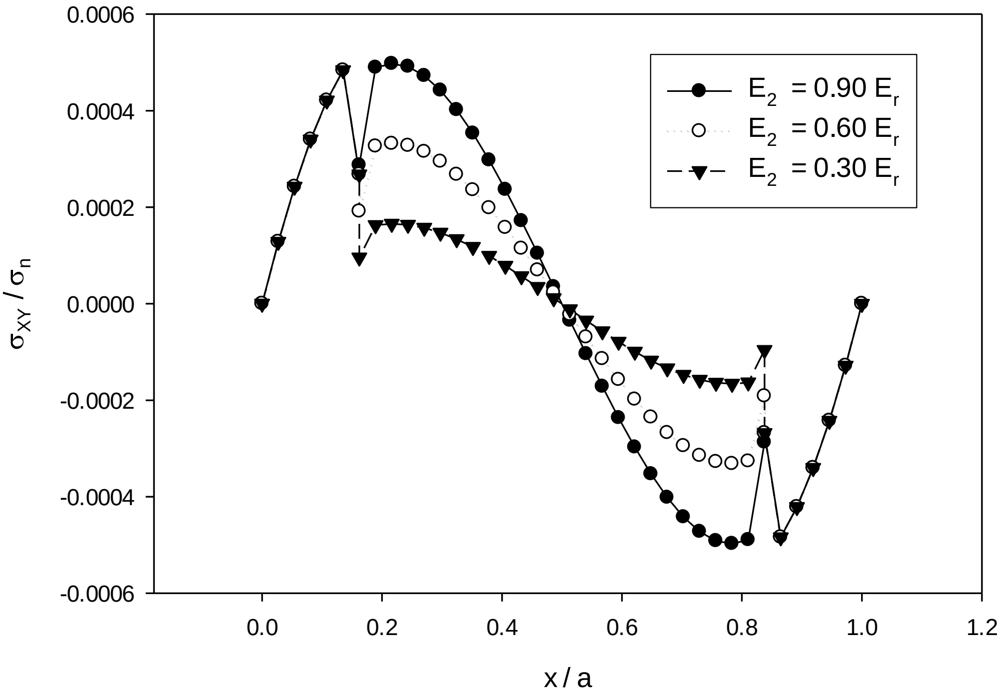


Figure 4.36: Comparison normalized shear stress (σ_{xy} / σ_n) distribution at different sections of a sandwich-structured composite.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

In this study a finite difference approach based on the displacement potential function formulation has been developed for the solution of two-dimensional elastic problems of sandwich-structured composite. The rectangular layers being perfectly bonded together to form a sandwich-structured composite are solved for stresses and displacement distributions in each layer by this approach. Since this approach deals with a single variable displacement potential function (ψ), it is found to be convenient to work with. An appropriate boundary management technique is followed to manage the interfacial boundary conditions. A programming code in the FOTRAN language is developed for this finite difference approach. Finite element results are obtained by using the commercial software. Results are presented in the graphs as non-dimensional form. Effects of Poisson's ratio and different loading condition are critically analyzed. Finally, the following conclusions are drawn in relation to the present research work:

1. The recently available methodology for the numerical solutions of mixed boundary-value elastic problem based on the ψ-formulation can be applied to the body of isotropic mechanical properties. Thus, a sandwich-structured composite of different mechanical properties in each layer cannot be solved by this available methodology. An extended and completely new computational approach of this ψ-formulation for sandwich-structured composite is presented in this thesis removing the limitations associated with the interfacial boundaries.
2. Completely new numerical formulations are developed in this thesis to solve the sandwich-structured composite problem. The numerical formulations with greater inclusion points at the interface provide better solution of the sandwich-structured composite as they ensure proper compatibility between the material layers.

3. The obtained solutions from the finite difference method are validated in numerous ways like comparison with the solution from finite element method, well-known published results and by qualitative intuition. Comparison between the results by the present finite difference method and the existing standard commercial software based on finite element method provide the good agreement. The computer program for sandwich-structure composite is also verified by keeping the same mechanical properties in each layer of the triple-layered material and obtaining the symmetrical results in each layer.
4. The stress and displacement distributions are presented in the present analysis for various combinations of mechanical properties and loadings. It is revealed that, when a sandwich-structured composite is subjected to mechanical loading, the corner zone as well as the interfacial zone is the most critical zone in engineering point of view. The stresses have a bumping effect at the interface due to different mechanical properties at the layers of the sandwich-structured composite.
5. To study the effect of modulus of elasticity, several problems of sandwich-structured composite are solved by FDM method keeping other properties and loading condition same. Here, the variation of modulus of elasticity in the materials has a significant effect on the distributions of stresses but less significant effect on the distribution of displacements.
6. In general, higher the modulus of elasticity causes higher normal and shear stresses in the material.

5.2 Recommendations for further research

The present study is perhaps the first attempt for the analysis of stresses by displacement potential function for two-dimensional elastic problems of sandwich-structured composite. The present formulation as well as the new computer program based on finite difference method has been developed to provide a new avenue for the investigations of two-dimensional elastic problems of sandwich-structured composite with all kinds of mixed boundary conditions of interest. In this connection, it is recommended that the program is further modified to incorporate the following improvements in the investigations of sandwich-structured composite problems.

- Attempts should be made to increase the total number of mesh points for more accurate results. Moreover, it will be appropriate to use smaller mesh size all along the boundary and the interface line. To incorporate the scheme a large coefficient matrix for a large number of unknowns has to be handled. It is, thus, suggested that the computer program should be organized in such a fashion that the partition of the matrix could be achieved by sub-dividing the coefficient matrix into four partitioned matrices.
- Experimental analysis could be performed to determine the stress and displacement of the sandwich-structured composite.
- The present computer program is applicable to double interface i.e. three layers in the composite material. A scrupulous effort could be made to develop a new computer program considering the same numerical formulations used in the present study. Thereby, it would be possible to solve multi-layered material by the new computer program and further, the rigorous effort could open a new horizon for three-dimensional analysis of the composite material.

REFERENCES

[1] Nath S.K.D., Ahmed S.R., “Displacement potential solution of stiffened composite struts subjected to eccentric loading” Applied Mathematical Modeling 33 (2009) 1761–1775

[2] Rijn V. L. P. V. M., “Towards the fastener less composite design”, Composites Part A: Applied Science and Manufacturing, Vol. 27, Issue 10, pp. 915-920, 1996.

[3] Hart-Smith, L. J., “Adhesive-Bonded Single-Lap Joints”, NASA-Langley Contract Report NASA-CR-112236, 1973.

[4] Wallis G. and Pomerantz, D. I., "Field Assisted Glass-Metal Sealing". Journal of Applied Physics 40 (10). pp. 3946–3949, 1969.

[5] P. Nitzsche and K. Klange and B. Schmidt and s. Grigull and U. Kreissig and B. Thoms and K. Herzog (1998). "Ion Drift Processes in Pyrex Type Alkali-Borosilicate Glass during Anodic Bonding". Electrochemical Society 145 (5). pp. 1755–1762.

[6] Dow, J. O., Jones, M. S. and Harwood, S. A., “A new approach to boundary modeling for finite difference applications in solid mechanics”, International Journal for Numerical Methods in Engineering, Vol.30, pp. 99-113, 1990.

[7] Hossain, M. Z., Ahmed, S. R. and Uddin, M. W., “An efficient algorithm for finite difference modeling of mixed-boundary-value elastic problems”, Advances in Engineering Software, Vol. 37, Issue 1, pp. 41-55, 2006.

[8] Runge, C., Z. Math. Physik., Vol. 56, pp. 225, 1908.

[9] Conway, H. D., Chow, L. and Morgan, G. W., “Analysis of deep beams”, Journal of applied mechanics Trans. ASME, Vol. 18, No. 2, pp. 163-172, 1951.

[10] Uddin, M. W., “Finite difference solution of two-dimensional elastic problems with mixed boundary conditions”, M. Sc. Thesis, Carleton University, Canada, 1966.

[11] Idris A.B.M, “A new approach to solution of mixed boundary value elastic problem”, M. Sc. Thesis, Bangladesh University of Engineering and Technology, Bangladesh, 1993.

[12] Ahmed, S. R., “Numerical solution of mixed boundary value elastic problems”, M. Sc. Thesis, Bangladesh University of Engineering and Technology, Bangladesh, 1993.

[13] Akanda, M. A. S., “Stress Analysis of gear teeth”, M. Sc. Thesis, Bangladesh University of Engineering and Technology, Bangladesh, 1997.

[14] Ahmed, S.R., Idris, A. B. M. and Uddin, M. W., “Numerical Solution of Both Ends Fixed Deep Beams”, Computers & Structures, Vol. 61, No. 1, pp. 21-29, 1996.

[15] Akanda, M. A. S., Ahmed, S.R., Khan, M. R., and Uddin, M. W., “A finite-difference scheme for mixed boundary-value problems of arbitrary-shaped elastic bodies”, Advances in Engineering Software, Vol. 31, No. 3, pp. 173-184, 2000.

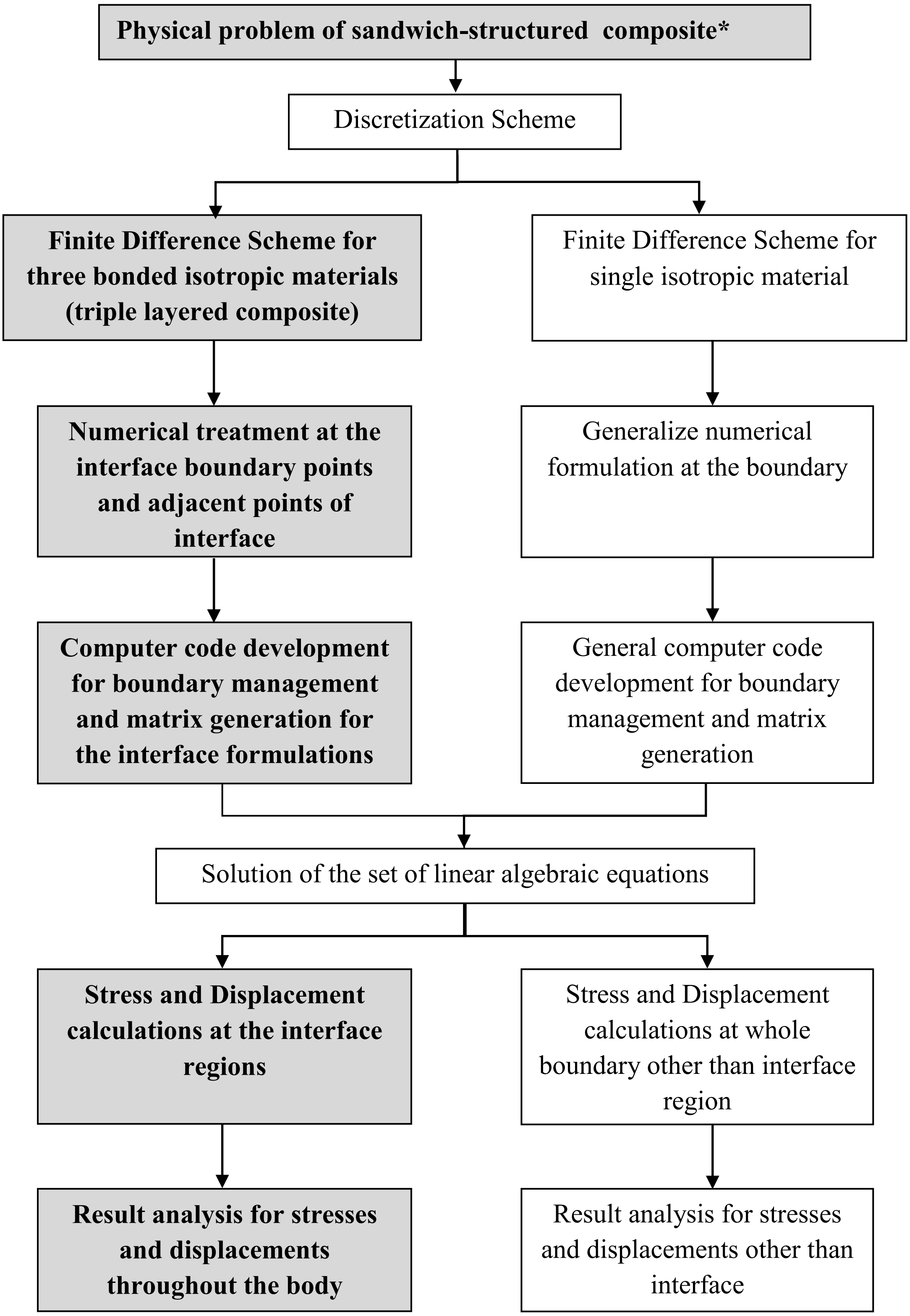
[16] Akanda, M. A. S., Ahmed, S.R. and Uddin, M. W., “Stress analysis of gear teeth using displacement potential function and finite-differences”, International Journal for Numerical Methods in Engineering, Vol-53, No. 7, pp 1629-1649, 2001.

[17] Hossain, M. Z., "A new approach to numerical solution of three dimensional mixed-boundary-value elastic problems", M. Sc. Thesis, Bangladesh University of Engineering and Technology, Bangladesh, 2004.
[18] Hossain, M. Z., Ahmed, S. R. and Uddin, M. W., "Generalized mathematical model for the solution of mixed-boundary-value elastic problems", Applied Mathematics and Computation, Vol. 169, Issue 2, pp. 1247-1275, 2005.
[19] Rahman, M. A., "A finite difference scheme for mixed boundary values of two dimensional elastic problems with internal hole", M. Sc. Thesis, Bangladesh University of Engineering and Technology, Bangladesh, 2006.
[20] Turner, M. J., Clough, R. W., Martin, H. C. and Topp, L. J., "Stiffness and deflection analysis of complex structures", J. aero. Sci, Vol. 23, pp. 805-823, 1956
[21] Clough, R. W., "The finite element in plane stress analysis", Proc. 2nd A.S.C.E Conf. on electronic Computation, Pitsburg, Sept. 1960.
[22] Zienkiewicz, O. C. and Cheung, Y. K., "The finite element method in structural and continuum mechanics", McGraw-Hill Publishing Company Ltd., England, 1967.
[23] Long L., Iwasaki S., Yin F. and Nagai K. "Prediction of Nominal Stress-Strain Curves of a Multi-Layered Composite Material by FE Analysis", Materials Transactions, Vol. 51, No. 12, pp. 2188-2195, 2010
[24] Sevecek O., Pletz M., Bermejo R., Supancic P. "Analytical Stress-Strain analysis of the laminates with orthotropic (isotropic) layers using Classical Laminate Theory", Materials Center Leoben Forschung GmbH, Presentation, 2011.
[25] Carreira R. P., Caron J. F., Diaz A. "Model of multilayered materials for interface stresses estimation and validation by finite element calculations", Materials Transactions, Vol. 51, No. 12, pp. 2188-2195, 2010.
[26] Wang Z., Wang W., Wang H. and Hu Y., "Stress Analysis on Layered Materials in Point Elastohydrodynamic-Lubricated Contacts", Tribology Letters (Springer), Vol. 35, Issue 3, pp. 229-244, 2009.
[27] Algbory A. M. R. M. "Stress analysis of the multi-layered thick cylinders", Al-Qadisiya Journal For Engineering Sciences, Vol. 4, No. 2, pp. 51-67, 2011.
[28] Arbaoui, J. Schmitt, Y., Pierrot, J.L., Royer F.X. "Numerical simulation and experimental bending behavior of multi-layered sandwich-structures", Journal of Theoretical and Applied Mechanics, 52, 2, pp. 431-442, Warsaw-2014
[29] Nath, S.K.D., Akanda, M.A.S., Ahmed, S.R. and Uddin, M.W., "Numerical investigation of bond-line stresses of tire tread section", International Journal of Applied Mechanics and Engineering, Vol. 13, No. 1, pp. 43-61, 2008.
[30] Alam, M. T., Akanda, M.A.S and Mandal, A.C., "Stress analysis of composite lamina having a square hole by finite difference technique", Proceedings of International Conference on Mechanical & Manufacturing Engineering, pp. 1-6, Johor Bahru, Malaysia, May 21-23, 2008.
[31] Bhuiyan, M. Y., "Interface Stress Analysis of Two Bonded Isotropic Materials by Finite Difference Method", M.Sc. Thesis, Bangladesh University of Engineering & Technology, Bangladesh, 2013.
[32] Timoshenko, S. and Goodier, J. N., Theory of Elasticity, 3rd Ed., McGraw-Hill Book Company, New York, N. Y., 1979.
[33] Sokolnikoff, I.S., Mathematical Theory of Elasticity, McGraw-Hill Book Company Inc., New York, 1956.

[34] Kirsch, G., VDI, Vol.42,pp.797-807,1898.

APPENDIX-A

Flowchart of the contribution of present study to the original main program



* Shaded portion indicates the contribution of present study.

APPENDIX-A

Input file: 1 (For boundary conditions and mechanical properties of the problem)

Boundary conditions and values

mm	lmn1	lmn2	bmn	rmn1	rmn2	q1	q2	q3	itoplf	ibotlf	ibotrg	itoprg	er1	er2
168	31	56	86	140	115	0.29	0.31	0.29	25	62	109	146	0.90	1.00

transfer & position controllers of the b.c.

j	i	ktb	kor	khv	kfld(i),i=1,mm)
1	24	4	1	1	1
25	25	4	1	1	1
26	31	1	1	2	1
32	43	1	1	2	2
44	44	1	2	2	2
45	56	1	2	2	2
57	61	1	2	2	3
62	62	4	2	1	3
63	86	4	2	1	3
87	108	4	3	1	3
109	109	4	3	1	3
110	112	4	3	2	3
113	114	4	3	2	3
115	115	4	3	2	2
116	126	4	3	2	2
127	127	4	3	2	2
128	129	4	4	2	2
129	139	4	4	2	2
140	140	4	4	2	1
141	142	4	4	2	1
143	145	4	4	2	1
146	146	4	4	1	1
147	168	4	4	1	1

non-zero boundary values (j,cnd1,cnd2)

147	-3.00E-04	0.00E+00
148	-3.00E-04	0.00E+00
149	-3.00E-04	0.00E+00
150	-3.00E-04	0.00E+00
151	-3.00E-04	0.00E+00
169	-0.000E-04	0.00E+00

TRANSFER & POSITION CONTROLLERS FOR ADDITIONAL B.C.

it(i)	kb(i)	kr(i)	kfd(i)	kh(i),i=1,ixces)
25	2	1	1	2
62	2	2	3	2
109	4	3	3	2
146	4	4	1	2

boundary values for additional points
(u v sigx sigy or sigxy anyone j=1,ixces)

0.0
0.0
0.0
0.0

Date revised 18052014

Boundary conditions and values

mm	lmn1	lmn2	bmn	rmn1	rmn2	q1	q2	q3	itoplf	ibotlf	ibotrg	itoprg	er1	er2
168	31	56	86	140	115	0.3	0.3	0.3	25	62	109	146	0.80	1.00

transfer & position controllers of the b.c.

j	i	ktb	kor	khv	kfld(i),i=1,mm)
1	24	4	1	1	1
25	25	4	1	1	1
26	31	1	1	2	1
32	43	1	1	2	2
44	44	1	2	2	2
45	56	1	2	2	2
57	61	1	2	2	3
62	62	4	2	1	3
63	86	4	2	1	3
87	108	4	3	1	3
109	109	4	3	1	3
110	112	4	3	2	3
113	114	4	3	2	3
115	115	4	3	2	2
116	126	4	3	2	2
127	127	4	3	2	2
128	129	4	4	2	2
129	139	4	4	2	2
140	140	4	4	2	1
141	142	4	4	2	1
143	145	4	4	2	1
146	146	4	4	1	1
147	168	4	4	1	1

non-zero boundary values (j,cnd1,cnd2)

110	3.00E-04	0.00E+00
111	3.00E-04	0.00E+00
112	3.00E-04	0.00E+00
113	3.00E-04	0.00E+00
114	3.00E-04	0.00E+00
115	3.00E-04	0.00E+00
116	3.00E-04	0.00E+00
117	3.00E-04	0.00E+00
118	3.00E-04	0.00E+00
119	3.00E-04	0.00E+00
120	3.00E-04	0.00E+00
121	3.00E-04	0.00E+00
122	3.00E-04	0.00E+00
123	3.00E-04	0.00E+00
124	3.00E-04	0.00E+00
125	3.00E-04	0.00E+00
126	3.00E-04	0.00E+00
127	3.00E-04	0.00E+00
128	3.00E-04	0.00E+00
129	3.00E-04	0.00E+00
130	3.00E-04	0.00E+00
131	3.00E-04	0.00E+00

132	3.00E-04	0.00E+00
133	3.00E-04	0.00E+00
134	3.00E-04	0.00E+00
135	3.00E-04	0.00E+00
136	3.00E-04	0.00E+00
137	3.00E-04	0.00E+00
138	3.00E-04	0.00E+00
139	3.00E-04	0.00E+00
140	3.00E-04	0.00E+00
141	3.00E-04	0.00E+00
142	3.00E-04	0.00E+00
143	3.00E-04	0.00E+00
144	3.00E-04	0.00E+00
145	3.00E-04	0.00E+00
169	0.000E-04	0.00E+00

TRANSFER & POSITION CONTROLLERS FOR ADDITIONAL B.C.

it(i)	kb(i)	kr(i)	kfd(i)	kh(i),i=1,ixces)
25	2	1	1	2
62	2	2	3	2
109	4	3	3	2
146	4	4	1	2

boundary values for additional points
(u v sigx sigy or sigxy anyone j=1,ixces)

0.0
0.0
3.00E-04
3.00E-04

Date revised 18052014
Boundary conditions and values

mm	lmn1	lmn2	bmn	rmn1	rmn2	q1	q2	q3	itoplf	ibotlf	ibotrg	itoprg	er1	er2
168	31	56	86	140	115	0.29	0.29	0.29	25	62	109	146	1.00	1.00

transfer & position controllers of the b.c.

j	i	ktb	kor	khv	kfld(i),i=1,mm)
1	24	4	1	1	1
25	25	4	1	1	1
26	31	1	1	2	1
32	43	1	1	2	2
44	44	1	2	2	2
45	56	1	2	2	2
57	61	1	2	2	3
62	62	4	2	1	3
63	86	4	2	1	3
87	108	4	3	1	3
109	109	4	3	1	3
110	112	4	3	2	3
113	114	4	3	2	3
115	115	4	3	2	2
116	126	4	3	2	2
127	127	4	3	2	2
128	129	4	4	2	2
129	139	4	4	2	2

140	140	4	4	2	1
141	142	4	4	2	1
143	145	4	4	2	1
146	146	4	4	1	1
147	168	4	4	1	1

non-zero boundary values (j,cnd1,cnd2)

147	-3.00E-04	0.00E+00
148	-3.00E-04	0.00E+00
149	-3.00E-04	0.00E+00
150	- 3.00E-04	0.00E+00
151	-3.00E-04	0.00E+00
169	-0.000E-04	0.00E+00

TRANSFER & POSITION CONTROLLERS FOR ADDITIONAL B.C.

it(i)	kb(i)	kr(i)	kfd(i)	kh(i),i=1,ixces)
25	2	1	1	2
62	2	2	3	2
109	4	3	3	2
146	4	4	1	2

boundary values for additional points
(u v sigx sigy or sigxy anyone j=1,ixces)
0.0
0.0
0.0
0.0

Date revised 18052014
Boundary conditions and values

mm	lmn1	lmn2	bmn	rmn1	rmn2	q1	q2	q3	itoplf	ibotlf	ibotrg	itoprg	er1	er2
168	31	56	86	140	115	0.29	0.31	0.29	25	62	109	146	0.90	1.00

transfer & position controllers of the b.c.

j	i	ktb	kor	khv	kfld(i),i=1,mm)
1	24	4	1	1	1
25	25	4	1	1	1
26	31	1	1	2	1
32	43	1	1	2	2
44	44	1	2	2	2
45	56	1	2	2	2
57	61	1	2	2	3
62	62	4	2	1	3
63	86	4	2	1	3
87	108	4	3	1	3
109	109	4	3	1	3
110	112	2	3	2	3
113	114	2	3	2	3
115	115	2	3	2	2
116	126	2	3	2	2
127	127	2	3	2	2
128	129	2	4	2	2
129	139	2	4	2	2
140	140	2	4	2	1
141	142	2	4	2	1
143	145	2	4	2	1

146 146 4 4 1 1
147 168 4 4 1 1

non-zero boundary values (j,cnd1,cnd2)

109	0.00E-04	0.00E+00
110	3.00E-04	0.00E+00
111	3.00E-04	0.00E+00
112	3.00E-04	0.00E+00
113	3.00E-04	0.00E+00
114	3.00E-04	0.00E+00
115	3.00E-04	0.00E+00
116	3.00E-04	0.00E+00
117	3.00E-04	0.00E+00
118	3.00E-04	0.00E+00
119	3.00E-04	0.00E+00
120	3.00E-04	0.00E+00
121	3.00E-04	0.00E+00
122	3.00E-04	0.00E+00
123	3.00E-04	0.00E+00
124	3.00E-04	0.00E+00
125	3.00E-04	0.00E+00
126	3.00E-04	0.00E+00
127	3.00E-04	0.00E+00
128	3.00E-04	0.00E+00
129	3.00E-04	0.00E+00
130	3.00E-04	0.00E+00
131	3.00E-04	0.00E+00
132	3.00E-04	0.00E+00
133	3.00E-04	0.00E+00
134	3.00E-04	0.00E+00
135	3.00E-04	0.00E+00
136	3.00E-04	0.00E+00
137	3.00E-04	0.00E+00
138	3.00E-04	0.00E+00
139	3.00E-04	0.00E+00
140	3.00E-04	0.00E+00
141	3.00E-04	0.00E+00
142	3.00E-04	0.00E+00
143	3.00E-04	0.00E+00
144	3.00E-04	0.00E+00
145	3.00E-04	0.00E+00
146	0.00E-04	0.00E+00
169	0.000E-04	0.00E+00

TRANSFER & POSITION CONTROLLERS FOR ADDITIONAL B.C.

it(i)	kb(i)	kr(i)	kfd(i)	kh(i),i=1,ixces)
25	2	1	1	2
62	2	2	3	2
109	2	3	3	2
146	2	4	1	2

boundary values for additional points
(u v sigx sigy or sigxy anyone j=1,ixces)

0.0
0.0
3.00E-04
3.00E-04

APPENDIX-A

Portion of the computer program for sandwich-structure formulation

```
  PROGRAM Main1
    PARAMETER(mtx=2500,net=100)
    Dimension a1(net,net)

    INTEGER a1,lyr

    real*8 r,h, x(mtx)
    real*8 uu(mtx),vv(mtx),ssgx(mtx),ssgy(mtx),ssgxy(mtx)
    REAL*8 uiu(mtx),viu(mtx),sxiu(mtx),syiu(mtx),sxyiu(mtx)
    REAL*8 uil(mtx),vil(mtx),sxil(mtx),syil(mtx),sxyil(mtx)
    REAL*8 uit(mtx),vit(mtx),sxit(mtx),syit(mtx),sxyit(mtx)
    REAL*8 uib(mtx),vib(mtx),sxib(mtx),syib(mtx),sxyib(mtx)

    common /o1a/a1
    common /o1b/lyr
    common /o2/r,h
    common /o3/kki,kkj,intfc1,intfc2,jdelams,jdelame
    common /o41a/nog
    COMMON /o42/x
    common /o44/uu,vv,ssgx,ssgy,ssgxy
    common /o45/uiu,viu,sxiu,syiu,sxyiu
    common /o46/uil,vil,sxil,syil,sxyil
    common /o47/uit,vit,sxit,syit,sxyit
    common /o48/uib,vib,sxib,syib,sxyib

    open(1,file='paramtr.txt',status='old')
    open(2,file='a3fld.txt',status='unknown')
    open(7,file='rslt090516x16.txt',status='unknown')
    open(8,file='si1.txt',status='unknown')

* initializing result parameters**
    do i=1,mtx
    uu(i)=0
    vv(i)=0
    ssgx(i)=0
    ssgy(i)=0
    ssgxy(i)=0
    uiu(i)=0
    viu(i)=0
    sxiu(i)=0
    syiu(i)=0
    sxyiu(i)=0
    uil(i)=0
    vil(i)=0
    sxil(i)=0
    syil(i)=0
    sxyil(i)=0
    uit(i)=0
    vit(i)=0
    sxit(i)=0
    syit(i)=0
    sxyit(i)=0
    uib(i)=0
    vib(i)=0
    sxib(i)=0
```

```
     syib(i)=0
     sxyib(i)=0
     end do

* field Generation
     read(1,*)   r, h, kki, kkj, intfc1, intfc2
     write(*,*)   r, h, kki, kkj, intfc1, intfc2

     WRITE(*,*) '     input value of lyr='
     read (*,*)lyr
* Initialize
     do i=1,kki
       do j=1,kkj
         a1(i,j)=0
       end do
     end do

************* a1 field node numbering *****************
* 1st line
       No=1
       do j=2,kkj-1
         a1(1,j)=No
         No=No+1
       END do

              do i=2,kki-1
               do j=1,kkj
                 a1(i,j)=No
                 No=No+1
               END do
             END do
              do j=2,kkj-1
                 a1(kki,j)=No
                 No=No+1
               END do
             nog=No-1

     GOTO 103


 103  WRITE(2,*)' nog    =', nog
     WRITE(2,*)' kki    =', kki
     WRITE(2,*)' kkj    =', kkj
     WRITE(2,*)' lyr    =', lyr

     do i=1,intfc1
     WRITE(2,11) (a1(i,j), j=1,kkj)
     end do
     do i=intfc1+1,intfc2
     WRITE(2,11) (a1(i,j), j=1,kkj)
     end do
     do i=intfc2+1,kki
     WRITE(2,11) (a1(i,j), j=1,kkj)
     end do

  11  FORMAT(9x,25i5)
     WRITE(*,*)'   run successful'
```

```
      write(*,*)'OK.  coef      is executing........'
      call coef

      write(*,*)'OK.  dcom      is executing........'
      call dcom

      write(*,*)'OK.  calc      is executing........'
      call calc

      do 125 i = 2, kki-1
  125 write(8,666) (x(a1(i,j)),j=2,kkj-1)

      WRITE(7,*)'uu'
      do 120 i = 2, kki-1
  120 write(7,666) (uu(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'vv'
      do 121 i = 2, kki-1
  121 write(7,666) (vv(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'ssgx'
      do 122 i = 2, kki-1
  122 write(7,666) (ssgx(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'ssgy'
      do 123 i = 2, kki-1
  123 write(7,666) (ssgy(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'ssgxy'
      do 124 i = 2, kki-1
  124 write(7,666) (ssgxy(a1(i,j)),j=2,kkj-1)

      i = intfc1
      WRITE(7,*)'uiu'
      write(7,666) (uiu(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'uil'
      write(7,666) (uil(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'viu'
      write(7,666) (viu(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'vil'
      write(7,666) (vil(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'sxiu'
      write(7,666) (sxiu(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'sxil'
      write(7,666) (sxil(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'syiu'
      write(7,666) (syiu(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'syil'
      write(7,666) (syil(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'sxyiu'
      write(7,666) (sxyiu(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'sxyil'
      write(7,666) (sxyil(a1(i,j)),j=2,kkj-1)

      i = intfc2
      WRITE(7,*)'uit'
      write(7,666) (uit(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'uib'
      write(7,666) (uib(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'vit'
      write(7,666) (vit(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'vib'
```

```
      write(7,666) (vib(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'sxit'
      write(7,666) (sxit(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'sxib'
      write(7,666) (sxib(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'syit'
      write(7,666) (syit(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'syib'
      write(7,666) (syib(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'sxyit'
      write(7,666) (sxyit(a1(i,j)),j=2,kkj-1)
      WRITE(7,*)'sxyib'
      write(7,666) (sxyib(a1(i,j)),j=2,kkj-1)


  666 format(1x, 50e12.4)

      stop
      END PROGRAM

*******************************************************************
      subroutine coef
*******************************************************************
      parameter (net=100, ibx=200,ics=10)
      parameter (mtx=2500)

      dimension a1(net,net)
      DIMENSION ktb(ibx),kor(ibx),khv(ibx),kfld(ibx),i1(ibx),j1(ibx)
      DIMENSION it(ics),kb(ics),kr(ics),kfd(ics),kh(ics),extrab(ics)
*       dimension i1(ibx),j1(ibx),i2(ibx),j2(ibx)

      integer a1,bmn,rmn1,rmn2,lyr

      character*1 head(65)

      REAL*8 sik(mtx,mtx),ck(mtx)
      real*8 zk1,zk2,zk3,zk4,zk5,q1,q2,q3,cnd1,cnd2,e,er1,er2,h
*       real*8 x1,x2,x3
      real*8 r,rs,c,c1,c2,p2
      real*8 u(ibx),v(ibx),sigx(ibx),sigy(ibx),sigxy(ibx)
      real*8 zk6,zk7,zk8,zk9,zk10,zk83,zk93
      REAL*8 e8,e9,e10,e11,e61,e62,e71,e72

      common /o1a/a1
      common /o1b/lyr
      common /o2/r,h
      common /o3/kki,kkj,intfc1,intfc2,jdelams,jdelame
      common /o4/mm /o19/i1,j1
      COMMON /o5/q1,q2,q3,e,er1,er2
      COMMON /o41/ sik, ck
      common /o41a/nog
      COMMON /o51/zk1,zk2,zk3,zk4,zk5,zk6,zk7,zk8,zk9,zk10,zk83,zk93
      COMMON /o58/e8,e9,e10,e11,e61,e62,e71,e72

      open(3,file='sikii.txt',status='unknown')
      open(4,file='in09516a.txt',status='old')
      open(5,file='inout3.txt',status='unknown')
      open(6,file='check.txt',status='unknown')
```

```
      WRITE(3,*)' nog    =', nog
      WRITE(3,*)' kki    =', kki
      WRITE(3,*)' kkj    =', kkj
      WRITE(3,*)' intfc1 =', intfc1
      WRITE(3,*)' intfc2 =', intfc2
      WRITE(3,*)' jdelams=', jdelams
      WRITE(3,*)' jdelame=', jdelame

*********************************************************************
*.................FOR THE POINTS AT BOUNDARY........................*
*********************************************************************
      read(4,53) (head(i),i=1,65)
      read(4,53) (head(i),i=1,65)
      read(4,53) (head(i),i=1,65)
      read(4,*)   mm,lmn1,lmn2,bmn,rmn1,rmn2,q1,q2,q3,itoplf,ibotlf,
     + ibotrg,itoprg,er1,er2
      write(5,52) mm,lmn1,lmn2,bmn,rmn1,rmn2,q1,q2,q3,itoplf,ibotlf,
     + ibotrg,itoprg,er1,er2
   52 format(1x,6i5,3f5.2,4i8,2f5.2)

      read(4,53) (head(i),i=1,65)
      read(4,53) (head(i),i=1,65)
      read(4,53) (head(i),i=1,65)
      write(5,53)(head(i),i=1,65)
   53 format(5x,65a1)

  32  read(4,*) j,i, ktb(j),kor(j),khv(j),kfld(j)
      write(5,59)j,i, ktb(j),kor(j),khv(j),kfld(j)
      IF(j .LT. i)then
         do j=j+1,i
         ktb(j)=ktb(j-1)
         kor(j)=kor(j-1)
         khv(j)=khv(j-1)
         kfld(j)=kfld(j-1)
         end do
      else
      endif
         IF(i .lt. mm)GOTO 32
   59 format(1x,6i6)

*initialize the boundary values
      do i=1,mm
        u(i)=0.0
        v(i)=0.0
        sigx(i)=0.0
        sigy(i)=0.0
        sigxy(i)=0.0
      end DO

      do 18 i=1,nog
       do 19 j=1,nog
         sik(i,j)=0.0
   19    continue
       ck(i)=0.0
   18 continue

* read & assign the non-zero boundary values started
```

```
      read(4,53) (head(i),i=1,65)
      read(4,53) (head(i),i=1,65)
      write(5,53)(head(i),i=1,65)

 33  read(4,*) j,cnd1,cnd2
      write(5,60) j,cnd1,cnd2
      l1=ktb(j)
      l3=khv(j)
      goto (34,35,36,37) l1
34   IF(l3 .EQ. 1)then
        u(j)=cnd1
        v(j)=cnd2
      else
        v(j)=cnd1
        u(j)=cnd2
      endif
      IF(j .le. mm)goto 33

35   IF(l3 .EQ. 1)then
        u(j)=cnd1
        sigxy(j)=cnd2
      else
        v(j)=cnd1
        sigxy(j)=cnd2
      endif
      IF(j .le. mm)goto 33

36   IF(l3 .EQ. 1)then
        sigx(j)=cnd1
        v(j)=cnd2
      else
        sigy(j)=cnd1
        u(j)=cnd2
      endif
      IF(j .le. mm)goto 33

37   IF(l3 .EQ. 1)then
        sigx(j)=cnd1
        sigxy(j)=cnd2
      else
        sigy(j)=cnd1
        sigxy(j)=cnd2
      endif
      IF(j .le. mm)goto 33
      WRITE(*,*)'   line 257'
  60 format(1x,i4,1x,2e12.4)
* read & assign the boundary values for excess points
      read(4,53) (head(i),i=1,65)
      read(4,53) (head(i),i=1,65)
      read(4,53) (head(i),i=1,65)
      write(5,53)(head(i),i=1,65)
      do i=1,4
        read(4,*) it(i),kb(i),kr(i),kfd(i),kh(i)
        write(5,59) it(i),kb(i),kr(i),kfd(i),kh(i)
      END do

      read(4,53) (head(i),i=1,65)
      read(4,53) (head(i),i=1,65)
```

```
      read(4,53) (head(i),i=1,65)
      write(5,53)(head(i),i=1,65)
      do i=1,4
       read(4,*) extrab(i)
       write(5,*) extrab(i)
      END do

* read & assign the non-zero boundary values finished
* i1 & j1 set

      k=itoplf+1
      do  i=1,itoplf
         i1(i)=2
         j1(i)=k
         k=k-1
      end do

      k=3
      do i=itoplf+1,ibotlf
         i1(i)=k
         j1(i)=2
         k=k+1
      end do

      k=3
      do  i=ibotlf+1, ibotrg
         i1(i)=kki-1
         j1(i)=k
         k=k+1
      end do

      k=kki-2
      do i=ibotrg+1,itoprg
         i1(i)=k
         j1(i)=kkj-1
         k=k-1
      end do

      k=kkj-2
      do  i=itoprg+1,mm
         i1(i)=2
         j1(i)=k
         k=k-1
      end do

       write(3,*) '    '
       write(3,*) '    i  i1  j1   '
      do i=1,mm
       write(3,13) i,i1(i),j1(i)
      end do
  13   format(1x,3i4)
* i1 j1 set finished*


*******************************************************************
*.................FOR THE POINTS on bounfary of THE BODY............*
*******************************************************************
```

```
* coefficient generation for boundary nodes started
*       IF (lyr .eq. 1) THEN
*        q2=q1
*        er=1.0
*     else
*     END IF

*          c21=(q1-1)/((1+q1)*r*r*h*h)
*          c22=(q2-1)/((1+q2)*r*r*h*h)
*          c23=(q3-1)/((1+q3)*r*r*h*h)
*          c31=(-2)/((1+q1)*h*h)
*          c32=(-2)/((1+q2)*h*h)
*          c33=(-2)/((1+q3)*h*h)
*          c61=(2+q1)/(2*r*h*h*h*(1+q1)*(1+q1))
*          c62=(2+q2)/(2*r*h*h*h*(1+q2)*(1+q2))
*          c63=(2+q3)/(2*r*h*h*h*(1+q3)*(1+q3))
*          c71=1/(2*r*r*r*h*h*h*(1+q1)*(1+q1))
*          c72=1/(2*r*r*r*h*h*h*(1+q2)*(1+q2))
*          c73=1/(2*r*r*r*h*h*h*(1+q3)*(1+q3))
*          c81=q1/((1+q1)*(1+q1)*2*r*r*h*h*h)
*          c82=q2/((1+q2)*(1+q2)*2*r*r*h*h*h)
*          c83=q3/((1+q3)*(1+q3)*2*r*r*h*h*h)
*          c91=1/(2*h*h*h*(1+q1)*(1+q1))
*          c92=1/(2*h*h*h*(1+q2)*(1+q2))
*          c93=1/(2*h*h*h*(1+q3)*(1+q3))

*          d31=-(6*c81)-c91
*          d41=(8*c81)+(3*c91)
*          d51=-(2*c81)-(3*c91)
*          d32=-(6*c82)-c92
*          d42=(8*c82)+(3*c92)
*          d52=-(2*c82)-(3*c92)
*          d33=-(6*c83)-c93
*          d43=(8*c83)+(3*c93)
*          d53=-(2*c83)-(3*c93)
             e8=(r*r)/(2.0*((q1+q2)/2.0))
        e9=(r*r)/(2.0*((q2+q3)/2.0))
        e10=(r*r*(2.0+((q1+q2)/2)))/2.0
        e11=(r*r*(2.0+((q2+q3)/2)))/2.0
        e61=((1-((q1+q2)/2.0))/2.0)/((1-((q1+q2)/2.0))+(2.0*r*r))
        e62=((1-((q2+q3)/2.0))/2.0)/((1-((q2+q3)/2.0))+(2.0*r*r))
        e71=(r*r)/((1-((q1+q2)/2.0))+(2.0*r*r))
        e72=(r*r)/((1-((q2+q3)/2.0))+(2.0*r*r))
        e81=(r*r*h*h*(1.0+((q1+q2)/2.0)))/(2.0*((1.0-((q1+q2)/2))
   ++(2.0*r*r)))
        e82=(r*r*h*h*(1.0+((q2+q3)/2.0)))/(2.0*((1.0-((q2+q3)/2))
   ++(2.0*r*r)))

     do  40 jj = 1, mm

          l1 = ktb(jj)
          l2 = kor(jj)
          l3 = khv(jj)
          l4 = kfld(jj)

*.......mu1 & mu2,  E1 & E2 assigned starts
     IF(l4 .EQ. 1)then
        c=q1
```

```
        e=1.0
     elseif(l4 .EQ. 2)then
        c=q2
        e=er1
     elseif(l4 .EQ. 3)then
        c=q3
        e=er2
     else
     endif
              c1=1.0-c
              c2=1.0+c

           zk6=(c1/2.0)/(c1+(2.0*r*r))
           zk7=(r*r)/(c1+(2.0*r*r))
           zk9=(r*r)/(2.0*c)
           zk10=(r*r*(2.0+c))/2.0

           zk8=(r*r*h*h*c2)/(2.0*(c1+(2.0*r*r)))
           p2=(c2*c2*r*r*r*h*h*h)/c
        zk83=c/(2*r*r*h*h*h*c2*c2)
        zk93=1/(2*h*h*h*c2*c2)

*.......mu1 & mu2,  E1 & E2 assigned  finished

     if (jj.eq.lmn1 .and. lyr .ne. 1) GOTO 112
     if (jj.eq.lmn2 .and. lyr .ne. 1) GOTO 212

     if (jj.eq.rmn2 .and. lyr .ne. 1) GOTO 213
     if (jj.eq.rmn1 .and. lyr .ne. 1) GOTO 113

****** for regular boundary formulas from now on **********

       if(l2 .eq. 1) then
          x1=1.0
          x2=1.0
          x3=1.0
          i0=1
          j0=1
       else if(l2 .eq. 2) then
          x1=-1.0
          x2=1.0
          x3=-1.0
          i0=-1
          j0=1
       else if(l2 .eq. 3) then
          x1=1.0
          x2=-1.0
          x3=-1.0
          i0=-1
          j0=-1
       else if(l2 .eq. 4) then
          x1=-1.0
          x2=-1.0
          x3=1.0
          i0=1
          j0=-1
       else
       endif
```

```
        i = i1(jj)
        j = j1(jj)

          m1 = a1(i,j)
          m2 = a1(i-i0,j)
          m3 = a1(i,j-j0)
          m4 = a1(i,j+j0)
          m5 = a1(i+i0,j)
          m6 = a1(i+i0,j+j0)
          m7 = a1(i,j+2*j0)
          m8 = a1(i+2*i0,j)
          m9 = a1(i+2*i0,j+j0)
          m10=a1(i+i0,j+2*j0)
          m11=a1(i+2*i0,j+2*j0)
          m12=a1(i-i0,j+j0)
          m13=a1(i-i0,j+2*j0)
          m14=a1(i,j+3*j0)
          m15=a1(i+i0,j-j0)
          m16=a1(i+2*i0,j-j0)
          m17=a1(i+3*i0,j)

*............................................................
      IF(l3 .EQ. 1)then

       goto (41,42,43,44),l1

 41      sik(m1,m1)=9.0
            sik(m1,m4)=-12.0
            sik(m1,m7)=3.0
            sik(m1,m5)=-12.0
            sik(m1,m6)=16.0
            sik(m1,m10)=-4.0
            sik(m1,m8)=3.0
            sik(m1,m9)=-4.0
            sik(m1,m11)=1.0
            ck(m1)=4.0*r*h*h*x1*u(jj)

        sik(m2,m2)=zk7
            sik(m2,m1)=-1.0
            sik(m2,m4)=zk6
            sik(m2,m3)=zk6
            sik(m2,m5)=zk7
            ck(m2)=-zk8*v(jj)
       goto 40

 42      sik(m1,m1)=9.0
            sik(m1,m4)=-12.0
            sik(m1,m7)=3.0
            sik(m1,m5)=-12.0
            sik(m1,m6)=16.0
            sik(m1,m10)=-4.0
            sik(m1,m8)=3.0
            sik(m1,m9)=-4.0
            sik(m1,m11)=1.0
            ck(m1)=4.0*r*h*h*x1*u(jj)
```

```
      sik(m2,m2)=(3.*zk9)
           sik(m2,m1)=3.-(10.*zk9)
           sik(m2,m3)=-1.5
           sik(m2,m4)=-1.5
           sik(m2,m5)=(12.*zk9)-4.
           sik(m2,m15)=2.
           sik(m2,m6)=2.
           sik(m2,m8)=1.-(6.*zk9)
           sik(m2,m16)=-0.5
           sik(m2,m9)=-0.5
           sik(m2,m17)=zk9
           ck(m2)=(p2/r)*x3*sigxy(jj)/e
     goto 40

43      sik(m1,m1)=((6.*zk9)-5.)
           sik(m1,m4)=(6.-(8.*zk9))
           sik(m1,m7)=((2.*zk9)-3.)
           sik(m1,m14)=0.5
           sik(m1,m3)=1.5
           sik(m1,m2)=-(3.*zk9)
           sik(m1,m12)=(4.*zk9)
           sik(m1,m13)=-zk9
           sik(m1,m5)=-(3.*zk9)
           sik(m1,m6)=(4.*zk9)
           sik(m1,m10)=-zk9
           ck(m1)=p2*x2*sigx(jj)/e

      sik(m2,m2)=zk7
           sik(m2,m1)=-1.0
           sik(m2,m4)=zk6
           sik(m2,m3)=zk6
           sik(m2,m5)=zk7
           ck(m2)=-zk8*v(jj)
     goto 40

44      sik(m1,m1)=((6.*zk9)-5.)
           sik(m1,m4)=(6.-(8.*zk9))
           sik(m1,m7)=((2.*zk9)-3.)
           sik(m1,m14)=0.5
           sik(m1,m3)=1.5
           sik(m1,m2)=-(3.*zk9)
           sik(m1,m12)=(4.*zk9)
           sik(m1,m13)=-zk9
           sik(m1,m5)=-(3.*zk9)
           sik(m1,m6)=(4.*zk9)
           sik(m1,m10)=-zk9
           ck(m1)=p2*x2*sigx(jj)/e

      sik(m2,m2)=3.*zk9
           sik(m2,m1)=3.-(10.*zk9)
           sik(m2,m3)=-1.5
           sik(m2,m4)=-1.5
           sik(m2,m5)=(12.*zk9)-4.
           sik(m2,m15)=2.
           sik(m2,m6)=2.
           sik(m2,m8)=1.-(6.*zk9)
           sik(m2,m16)=-0.5
           sik(m2,m9)=-0.5
```

```
            sik(m2,m17)=zk9
            ck(m2)=(p2/r)*x3*sigxy(jj)/e

      ELSEIF(l3 .EQ. 2)then

      goto (45,46,47,48),l1

45      sik(m3,m2)=zk7
            sik(m3,m1)=-1.0
            sik(m3,m4)=zk6
            sik(m3,m3)=zk6
            sik(m3,m5)=zk7
            ck(m3)=-zk8*v(jj)

        sik(m1,m1)=9.0
            sik(m1,m4)=-12.0
            sik(m1,m7)=3.0
            sik(m1,m5)=-12.0
            sik(m1,m6)=16.0
            sik(m1,m10)=-4.0
            sik(m1,m8)=3.0
            sik(m1,m9)=-4.0
            sik(m1,m11)=1.0
            ck(m1)=4.0*r*h*h*x1*u(jj)
      goto 40

46      sik(m3,m2)=zk7
            sik(m3,m1)=-1.0
            sik(m3,m4)=zk6
            sik(m3,m3)=zk6
            sik(m3,m5)=zk7
            ck(m3)=-zk8*v(jj)

        sik(m1,m2)=3.*zk9
            sik(m1,m1)=3.-(10.*zk9)
            sik(m1,m3)=-1.5
            sik(m1,m4)=-1.5
            sik(m1,m5)=(12.*zk9)-4.
            sik(m1,m15)=2.
            sik(m1,m6)=2.
            sik(m1,m8)=1.-(6.*zk9)
            sik(m1,m16)=-0.5
            sik(m1,m9)=-0.5
            sik(m1,m17)=zk9
            ck(m1)=(p2/r)*x3*sigxy(jj)/e
      goto 40

47      sik(m3,m3)=-1.5
            sik(m3,m1)=5.0+(6.0*zk10)
            sik(m3,m4)=-(6.+(8.*zk10))
            sik(m3,m7)=(3.0+(2.*zk10))
            sik(m3,m14)=-0.5
            sik(m3,m2)=-(3.*zk10)
            sik(m3,m12)=(4.*zk10)
            sik(m3,m13)=-zk10
            sik(m3,m5)=-(3.*zk10)
            sik(m3,m6)=(4.*zk10)
            sik(m3,m10)=-zk10
```

```
          ck(m3)=-c*p2*x2*sigy(jj)/e

       sik(m1,m1)=9.0
          sik(m1,m4)=-12.0
          sik(m1,m7)=3.0
          sik(m1,m5)=-12.0
          sik(m1,m6)=16.0
          sik(m1,m10)=-4.0
          sik(m1,m8)=3.0
          sik(m1,m9)=-4.0
          sik(m1,m11)=1.0
          ck(m1)=4.0*r*h*h*x1*u(jj)
     goto 40

48      sik(m3,m3)=-1.5
          sik(m3,m1)=(5.0+(6.0*zk10))
          sik(m3,m4)=-(6.+(8.*zk10))
          sik(m3,m7)=(3.0+(2.*zk10))
          sik(m3,m14)=-0.5
          sik(m3,m2)=-(3.*zk10)
          sik(m3,m12)=(4.*zk10)
          sik(m3,m13)=-zk10
          sik(m3,m5)=-(3.*zk10)
          sik(m3,m6)=(4.*zk10)
          sik(m3,m10)=-zk10
          ck(m3)=-c*p2*x2*sigy(jj)/e

       sik(m1,m2)=(3.*zk9)
          sik(m1,m1)=(3.-(10.*zk9))
          sik(m1,m3)=-1.5
          sik(m1,m4)=-1.5
          sik(m1,m5)=((12.*zk9)-4.)
          sik(m1,m15)=2.
          sik(m1,m6)=2.
          sik(m1,m8)=(1.-(6.*zk9))
          sik(m1,m16)=-0.5
          sik(m1,m9)=-0.5
          sik(m1,m17)=zk9
          ck(m1)=(p2/r)*x3*sigxy(jj)/e

     else
     endif

     WRITE(6,10)jj,l1,l2,l3,l4,m1,m2,m3,zk7,zk8,zk9
  10 FORMAT(1x,8i5,1x,3f10.7)

      GOTO 40

112     i = i1(jj)
       j = j1(jj)

         x1=1.0
         x2=1.0
         x3=1.0
         i0=1
         j0=1

          m1 = a1(i,j)
```

```
        m2 = a1(i-i0,j)
        m3 = a1(i,j-j0)
        m4 = a1(i,j+j0)
        m5 = a1(i+i0,j)
        m6 = a1(i+i0,j+j0)
        m7 = a1(i,j+2*j0)
        m8 = a1(i+2*i0,j)
        m9 = a1(i+2*i0,j+j0)
        m10=a1(i+i0,j+2*j0)
        m11=a1(i+2*i0,j+2*j0)
        m12=a1(i-i0,j+j0)
        m13=a1(i-i0,j+2*j0)
        m14=a1(i,j+3*j0)
        m15=a1(i+i0,j-j0)
        m16=a1(i+2*i0,j-j0)
        m17=a1(i+3*i0,j)

      sik(m3,m2)=e71
          sik(m3,m1)=-1.0
          sik(m3,m4)=e61
          sik(m3,m3)=e61
          sik(m3,m5)=e71
          ck(m3)=-e81*v(jj)

      sik(m1,m1)=9.0
          sik(m1,m4)=-12.0
          sik(m1,m7)=3.0
          sik(m1,m5)=-12.0
          sik(m1,m6)=16.0
          sik(m1,m10)=-4.0
          sik(m1,m8)=3.0
          sik(m1,m9)=-4.0
          sik(m1,m11)=1.0
          ck(m1)=4.0*r*h*h*x1*u(jj)

    IF(jj .EQ. 18) WRITE(*,*)m1,m3

      GOTO 40

212    i = i1(jj)
     j = j1(jj)

       x1=-1.0
       x2=1.0
       x3=-1.0
       i0=-1
       j0=1

        m1 = a1(i,j)
        m2 = a1(i-i0,j)
        m3 = a1(i,j-j0)
        m4 = a1(i,j+j0)
        m5 = a1(i+i0,j)
        m6 = a1(i+i0,j+j0)
        m7 = a1(i,j+2*j0)
        m8 = a1(i+2*i0,j)
        m9 = a1(i+2*i0,j+j0)
        m10=a1(i+i0,j+2*j0)
```

```
        m11=a1(i+2*i0,j+2*j0)
        m12=a1(i-i0,j+j0)
        m13=a1(i-i0,j+2*j0)
        m14=a1(i,j+3*j0)
        m15=a1(i+i0,j-j0)
        m16=a1(i+2*i0,j-j0)
        m17=a1(i+3*i0,j)

      sik(m3,m2)=e72
          sik(m3,m1)=-1.0
          sik(m3,m4)=e62
          sik(m3,m3)=e62
          sik(m3,m5)=e72
          ck(m3)=-e82*v(jj)

      sik(m1,m1)=9.0
          sik(m1,m4)=-12.0
          sik(m1,m7)=3.0
          sik(m1,m5)=-12.0
          sik(m1,m6)=16.0
          sik(m1,m10)=-4.0
          sik(m1,m8)=3.0
          sik(m1,m9)=-4.0
          sik(m1,m11)=1.0
          ck(m1)=4.0*r*h*h*x1*u(jj)

    IF(jj .EQ. 18) WRITE(*,*)m1,m3

      GOTO 40

213   i = i1(jj)
    j = j1(jj)

       x1=-1.0
       x2=-1.0
       x3=1.0
       i0=1
       j0=-1

        m1 = a1(i,j)
        m2 = a1(i-i0,j)
        m3 = a1(i,j-j0)
        m4 = a1(i,j+j0)
        m5 = a1(i+i0,j)
        m6 = a1(i+i0,j+j0)
        m7 = a1(i,j+2*j0)
        m8 = a1(i+2*i0,j)
        m9 = a1(i+2*i0,j+j0)
        m10=a1(i+i0,j+2*j0)
        m11=a1(i+2*i0,j+2*j0)
        m12=a1(i-i0,j+j0)
        m13=a1(i-i0,j+2*j0)
        m14=a1(i,j+3*j0)
        m15=a1(i+i0,j-j0)
        m16=a1(i+2*i0,j-j0)
        m17=a1(i+3*i0,j)

      sik(m3,m3)=-1.5
```

```
          sik(m3,m1)=(5.0+(6.0*e11))
          sik(m3,m4)=-(6.+(8.*e11))
          sik(m3,m7)=(3.0+(2.*e11))
          sik(m3,m14)=-0.5
          sik(m3,m2)=-(3.*e11)
          sik(m3,m12)=(4.*e11)
          sik(m3,m13)=-e11
          sik(m3,m5)=-(3.*e11)
          sik(m3,m6)=(4.*e11)
          sik(m3,m10)=-e11
          ck(m3)=-c*p2*x2*sigy(jj)/e

      sik(m1,m2)=(3.*e9)
          sik(m1,m1)=(3.-(10.*e9))
          sik(m1,m3)=-1.5
          sik(m1,m4)=-1.5
          sik(m1,m5)=((12.*e9)-4.)
          sik(m1,m15)=2.
          sik(m1,m6)=2.
          sik(m1,m8)=(1.-(6.*e9))
          sik(m1,m16)=-0.5
          sik(m1,m9)=-0.5
          sik(m1,m17)=e9
          ck(m1)=(p2/r)*x3*sigxy(jj)/e

    IF(jj .EQ. 52) WRITE(*,*)m1,m3

      GOTO 40

113    i = i1(jj)
     j = j1(jj)

       x1=1.0
       x2=-1.0
       x3=-1.0
       i0=-1
       j0=-1

        m1 = a1(i,j)
        m2 = a1(i-i0,j)
        m3 = a1(i,j-j0)
        m4 = a1(i,j+j0)
        m5 = a1(i+i0,j)
        m6 = a1(i+i0,j+j0)
        m7 = a1(i,j+2*j0)
        m8 = a1(i+2*i0,j)
        m9 = a1(i+2*i0,j+j0)
        m10=a1(i+i0,j+2*j0)
        m11=a1(i+2*i0,j+2*j0)
        m12=a1(i-i0,j+j0)
        m13=a1(i-i0,j+2*j0)
        m14=a1(i,j+3*j0)
        m15=a1(i+i0,j-j0)
        m16=a1(i+2*i0,j-j0)
        m17=a1(i+3*i0,j)

      sik(m3,m3)=-1.5
          sik(m3,m1)=(5.0+(6.0*e10))
```

```
              sik(m3,m4)=-(6.+(8.*e10))
              sik(m3,m7)=(3.0+(2.*e10))
              sik(m3,m14)=-0.5
              sik(m3,m2)=-(3.*e10)
              sik(m3,m12)=(4.*e10)
              sik(m3,m13)=-e10
              sik(m3,m5)=-(3.*e10)
              sik(m3,m6)=(4.*e10)
              sik(m3,m10)=-e10
              ck(m3)=-c*p2*x2*sigy(jj)/e

           sik(m1,m2)=(3.*e8)
              sik(m1,m1)=(3.-(10.*e8))
              sik(m1,m3)=-1.5
              sik(m1,m4)=-1.5
              sik(m1,m5)=((12.*e8)-4.)
              sik(m1,m15)=2.
              sik(m1,m6)=2.
              sik(m1,m8)=(1.-(6.*e8))
              sik(m1,m16)=-0.5
              sik(m1,m9)=-0.5
              sik(m1,m17)=e8
              ck(m1)=(p2/r)*x3*sigxy(jj)/e

      IF(jj .EQ. 52) WRITE(*,*)m1,m3


 40   continue
*********************************************************************
*.................FOR THE POINTS INSIDE THE BODY....................*
*********************************************************************
             rs = r*r
              zk1=rs*rs
              zk2=4.0*(rs*rs+rs)
              zk3=4.0*(1.0+rs)
              zk4=6.0+(8.0*rs)+(6.0*rs*rs)
              zk5=2.0*rs

* Gov Eqn Formula Loop

      GOTO 104
 104   do 93 i=3, kki-2
        do 94 j= 3, kkj-2

           m1 = a1(i,j)
           m2 = a1(i,j+1)
           m3 = a1(i,j-1)
           m4 = a1(i+1,j)
           m5 = a1(i-1,j)
           m6 = a1(i+1,j+1)
           m7 = a1(i-1,j+1)
           m8 = a1(i-1,j-1)
           m9 = a1(i+1,j-1)
           m10 = a1(i,j+2)
           m11 = a1(i-2,j)
           m12 = a1(i,j-2)
           m13 = a1(i+2,j)
```

```
          sik(m1,m1) =zk4
          sik(m1,m2) =-zk3
          sik(m1,m3) =-zk3
          sik(m1,m4) =-zk2
          sik(m1,m5) =-zk2
          sik(m1,m6) =zk5
          sik(m1,m7) =zk5
          sik(m1,m8) =zk5
          sik(m1,m9) =zk5
          sik(m1,m10) =1.0
          sik(m1,m11) =zk1
          sik(m1,m12) =1.0
          sik(m1,m13) =zk1
          ck(m1) =0.0

94    continue
93   continue
    write (*,*) '  line no. 1101', lyr
    GOTO 106

106    do 70 jj = 1, 4
           l5 = kb(jj)
           l6 = kr(jj)
           l7 = kfd(jj)
           l8 = kh(jj)

      if(l6 .eq. 1) then
         x1=1.0
         x2=1.0
         x3=1.0
         i0=1
         j0=1
      else if(l6 .eq. 2) then
         x1=-1.0
         x2=1.0
         x3=-1.0
         i0=-1
         j0=1
      else if(l6 .eq. 3) then
         x1=1.0
         x2=-1.0
         x3=-1.0
         i0=-1
         j0=-1
      else if(l6 .eq. 4) then
         x1=-1.0
         x2=-1.0
         x3=1.0
         i0=1
         j0=-1
      else
      endif

    IF (l7 .eq. 1) THEN
       c=q1
       e=1.0
    elseif(l7 .EQ. 2)then
       c=q2
```

```
      e=er1
    elseif(l7 .EQ. 3)then
      c=q3
      e=er2
    else
    endif
              c1=1.0-c
              c2=1.0+c

          zk6=(c1/2.0)/(c1+(2.0*r*r))
          zk7=(r*r)/(c1+(2.0*r*r))
          zk9=(r*r)/(2.0*c)
          zk10=(r*r*(2.0+c))/2.0

          zk8=(r*r*h*h*c2)/(2.0*(c1+(2.0*r*r)))
          p2=(c2*c2*r*r*r*h*h*h)/c

      i = i1(it(jj))
      j = j1(it(jj))

        m1 = a1(i,j)
        m2 = a1(i-i0,j)
        m3 = a1(i,j-j0)
        m4 = a1(i,j+j0)
        m5 = a1(i+i0,j)
        m6 = a1(i+i0,j+j0)
        m7 = a1(i,j+2*j0)
        m8 = a1(i+2*i0,j)
        m9 = a1(i+2*i0,j+j0)
        m10=a1(i+i0,j+2*j0)
        m11=a1(i+2*i0,j+2*j0)
        m12=a1(i-i0,j+j0)
        m13=a1(i-i0,j+2*j0)
        m14=a1(i,j+3*j0)
        m15=a1(i+i0,j-j0)
        m16=a1(i+2*i0,j-j0)
        m17=a1(i+3*i0,j)

    IF(l8 .EQ. 1)then
      me = m2
    else
      me = m3
    endif
*...........................................................
    goto (71,72,73,74,75),l5

 71     sik(me,m1)=9*x1
          sik(me,m4)=-12*x1
          sik(me,m7)=3*x1
          sik(me,m5)=-12*x1
          sik(me,m6)=16*x1
          sik(me,m10)=-4*x1
          sik(me,m8)=3*x1
          sik(me,m9)=-4*x1
          sik(me,m11)=1*x1
          ck(me)=4*r*h*h*extrab(jj)
     goto 70
 72      sik(me,m2)=zk7
```

```
            sik(me,m1)=-1.0
            sik(me,m4)=zk6
            sik(me,m3)=zk6
            sik(me,m5)=zk7
            ck(me)=-zk8*extrab(jj)
      goto 70
 73      sik(me,m1)=(6*zk9-5)
            sik(me,m4)=(6-8*zk9)
            sik(me,m7)=(2*zk9-3)
            sik(me,m14)=0.5
            sik(me,m3)=1.5
            sik(me,m2)=-3*zk9
            sik(me,m12)=4*zk9
            sik(me,m13)=-zk9
            sik(me,m5)=-3*zk9
            sik(me,m6)=4*zk9
            sik(me,m10)=-zk9
            ck(me)=p2*x2*extrab(jj)
      goto 70
 74      sik(me,m3)=-1.5
            sik(me,m1)=(5+6*zk10)
            sik(me,m4)=-(6+8*zk10)
            sik(me,m7)=(3+2*zk10)
            sik(me,m14)=-0.5
            sik(me,m2)=-3*zk10
            sik(me,m12)=4*zk10
            sik(me,m13)=-zk10
            sik(me,m5)=-3*zk10
            sik(me,m6)=4*zk10
            sik(me,m10)=-zk10
            ck(me)=-c*p2*x2*extrab(jj)
      goto 70
 75       sik(me,m2)=3*zk9
            sik(me,m1)=(3-10*zk9)
            sik(me,m3)=-1.5
            sik(me,m4)=-1.5
            sik(me,m5)=(12*zk9-4)
            sik(me,m15)=2
            sik(me,m6)=2
            sik(me,m8)=(1-6*zk9)
            sik(me,m16)=-0.5
            sik(me,m9)=-0.5
            sik(me,m17)=zk9
            ck(me)=(p2/r)*x3*extrab(jj)
 70   continue
* for excess point finished

******************************************
      if (lyr.eq.3) GOTO 107


 107  WRITE(*,*)'    line= 1251'

     WRITE(3,*)'     nog =    ', nog
     do 28 i =1, nog
        write(3,12) sik(i,i), ck(i), i
 28  continue
 12  FORMAT(1x, 2e12.5, i4)
```

```
*************************************************************
*   SUBROUTINE "DCOM" FOR THE DECOMPOSITION INTO LOWER AND   *
*   UPPER TRAINGULAR MATRICES TO GET SOLUTION              *
*************************************************************
      subroutine dcom

      parameter (mtx=2500)
      real*8  a(mtx,mtx),al(mtx,mtx),au(mtx,mtx),z(mtx),x(mtx),b(mtx)
      common /o41/a,b
      common /o41a/n
      common /o42/x

      do 50 i=1,n
      x(i)=0.0
      z(i)=0.0
      do 50 j=1,n
      al(i,j)=0.0
      au(i,j)=0.0
50    continue
      do 60 i=1,n
      au(i,i)=1.0
      al(i,1)=a(i,1)
60    au(1,i)=a(1,i)/al(1,1)
      do 10 j=2,n
      do 20 i=j,n
      sum=0.0
      do 18 k=1,(j-1)
18    sum=sum+al(i,k)*au(k,j)
20    al(i,j)=a(i,j)-sum
      if (j .eq. n) go to 10
      do 40 jj=(j+1),n
      sum=0.0
      do 70 kk=1,(j-1)
70    sum=sum+al(j,kk)*au(kk,jj)
40    au(j,jj)=(a(j,jj)-sum)/al(j,j)
10    continue
      z(1)=b(1)/al(1,1)
      do 80 i=2,n
      sum=0.0
      do 90 k=1,(i-1)
90    sum=sum+al(i,k)*z(k)
80    z(i)=(b(i)-sum)/al(i,i)
      x(n)=z(n)
      do 100 i=2,n
      l=n-i+1
      sum=0.0
      do 110 k=(l+1),n
110   sum=sum+au(l,k)*x(k)
100   x(l)=z(l)-sum
      return
      end
****************************************************************
      subroutine calc
****************************************************************
      parameter (net=100, ibx=200)
      parameter (mtx=2500)
```

```
      dimension a1(net,net)
      dimension i1(ibx),j1(ibx)

      integer a1,lyr

      real*8 x1,x2,x3
      real*8 h,r,c,c1,c2,p2
      real*8 si(mtx)
      real*8 uu(mtx),vv(mtx),ssgx(mtx),ssgy(mtx),ssgxy(mtx)

*      REAL*8 c21,c22,c23,c31,c32,c33,e8,e9,e10,e11,e61,e62,e71,e72
      REAL*8 c121,c131,c141,c151,c161,c171,c181,c191,d131,d141,d151
      REAL*8 c122,c132,c142,c152,c162,c172,c182,c192,d132,d142,d152
      REAL*8 c123,c133,c143,c153,c163,c173,c183,c193,d133,d143,d153

      REAL*8 uiu(mtx),viu(mtx),sxiu(mtx),syiu(mtx),sxyiu(mtx)
      REAL*8 uil(mtx),vil(mtx),sxil(mtx),syil(mtx),sxyil(mtx)
      REAL*8 uit(mtx),vit(mtx),sxit(mtx),syit(mtx),sxyit(mtx)
      REAL*8 uib(mtx),vib(mtx),sxib(mtx),syib(mtx),sxyib(mtx)

      real*8 zk1,zk2,zk3,zk4,zk5,zk6,zk7,zk8,zk9,zk10,zk83,zk93
      REAL*8 q1,q2,q3, e, er1,er2
      REAL*8 e8,e9,e10,e11,e61,e62,e71,e72

      common /o1a/a1
      common /o1b/lyr
      common /o2/r,h
      common /o3/kki,kkj,intfc1,intfc2,jdelams,jdelame
      common /o4/mm /o19/i1,j1
      COMMON /o5/q1,q2,q3,e,er1,er2
      COMMON /o51/zk1,zk2,zk3,zk4,zk5,zk6,zk7,zk8,zk9,zk10,zk83,zk93
      COMMON /o58/e8,e9,e10,e11,e61,e62,e71,e72

      COMMON /o41/ sik, ck

      common /o41a/nog
      common /o42/si
      common /o44/uu,vv,ssgx,ssgy,ssgxy
      common /o45/uiu,viu,sxiu,syiu,sxyiu
      common /o46/uil,vil,sxil,syil,sxyil
      common /o47/uit,vit,sxit,syit,sxyit
      common /o48/uib,vib,sxib,syib,sxyib

*****************************************
      write (*,*)'lyr=  ',lyr
      GOTO (208,209,210) lyr

  210      iia=intfc1/2
          jja=kkj/2


*      upper field calculation started
          e=1.0
          c=q1
          c1=1.0-c
          c2=1.0+c
               zk6=(c1/2.0)/(c1+(2.0*r*r))
               zk7=(r*r)/(c1+(2.0*r*r))
```

```
            zk9=(r*r)/(2.0*c)
            zk10=(r*r*(2.0+c))/2.0

            zk8=(r*r*h*h*c2)/(2.0*(c1+(2.0*r*r)))
            p2=(c2*c2*r*r*r*h*h*h)/c

      do 91 i = 2, intfc1-1
       do 92 j = 2, kkj-1

       IF(i .LE. iia .and. j .le. jja)then
          x1=1.0
          x2=1.0
          x3=1.0
          i0=1
          j0=1
       else IF(i .gt. iia .and. j .le. jja)then
          x1=-1.0
          x2=1.0
          x3=-1.0
          i0=-1
          j0=1
       else IF(i .gt. iia .and. j .gt. jja)then
          x1=1.0
          x2=-1.0
          x3=-1.0
          i0=-1
          j0=-1
       else IF(i .le. iia .and. j .gt. jja)then
          x1=-1.0
          x2=-1.0
          x3=1.0
          i0=1
          j0=-1
       else
       endif

          m1 = a1(i,j)
          m2 = a1(i-i0,j)
          m3 = a1(i,j-j0)
          m4 = a1(i,j+j0)
          m5 = a1(i+i0,j)
          m6 = a1(i+i0,j+j0)
          m7 = a1(i,j+2*j0)
          m8 = a1(i+2*i0,j)
          m9 = a1(i+2*i0,j+j0)
          m10=a1(i+i0,j+2*j0)
          m11=a1(i+2*i0,j+2*j0)
          m12=a1(i-i0,j+j0)
          m13=a1(i-i0,j+2*j0)
          m14=a1(i,j+3*j0)
          m15=a1(i+i0,j-j0)
          m16=a1(i+2*i0,j-j0)
          m17=a1(i+3*i0,j)
*..........................................................
      uu(m1)=(9.0*si(m1)-12.0*si(m4)+3.0*si(m7)-12.0*si(m5)+16.0*si(m6)
     +-4.0*si(m10)+3.0*si(m8)-4.0*si(m9)+si(m11))/(4.0*r*h*h*x1)

      vv(m1)=(zk7*(si(m2)+si(m5))+zk6*(si(m3)+si(m4))-si(m1))/(-zk8)
```

```
      ssgx(m1)=(si(m1)*(6.*zk9-5.)+si(m4)*(6.-8.*zk9)+
     +si(m7)*(2.*zk9-3.)+si(m14)*0.5+si(m3)*1.5+si(m2)*(-3.*zk9)+
     +si(m12)*4.*zk9+si(m13)*(-zk9)+si(m5)*(-3.*zk9)+si(m6)*4.*zk9+
     +si(m10)*(-zk9))*e/(p2*x2)

      ssgy(m1)=(si(m3)*(-1.5)+si(m1)*(5.0+6.0*zk10)-si(m4)*(6.+8.*zk10)
     ++si(m7)*(3.0+2.*zk10)-si(m14)*0.5-si(m2)*3.*zk10+si(m12)*4.*zk10
     +-si(m13)*zk10-si(m5)*3.*zk10+si(m6)*4.*zk10
     +-si(m10)*zk10)*e/(-c*p2*x2)

      ssgxy(m1)=(si(m2)*3.*zk9+si(m1)*(3.-10.*zk9)-si(m3)*1.5-
     +si(m4)*1.5+si(m5)*(12.*zk9-4.)+si(m15)*2.+si(m6)*2.+
     +si(m8)*(1.-6.*zk9)-si(m16)*0.5-
     +si(m9)*0.5+si(m17)*zk9)*(r*e/(p2*x3))
   92   continue
   91  continue
*  upper field finished


*      upper interface calculation started
*           iid=intfc1
          jjd=kkj/2
          e=(1.0+er1)/2
          c=(q1+q2)/2
          c1=1.0-c
          c2=1.0+c
               zk6=(c1/2.0)/(c1+(2.0*r*r))
               zk7=(r*r)/(c1+(2.0*r*r))
               zk9=(r*r)/(2.0*c)
               zk10=(r*r*(2.0+c))/2.0

               zk8=(r*r*h*h*c2)/(2.0*(c1+(2.0*r*r)))
               p2=(c2*c2*r*r*r*h*h*h)/c
               e8=(r*r)/(2.0*((q1+q2)/2.0))
*         e9=(r*r)/(2.0*((q2+q3)/2.0))
         e10=(r*r*(2.0+((q1+q2)/2)))/2.0
*         e11=(r*r*(2.0+((q2+q3)/2)))/2.0
          e61=((1-((q1+q2)/2.0))/2.0)/((1-((q1+q2)/2.0))+(2.0*r*r))
*           e62=((1-((q2+q3)/2.0))/2.0)/((1-((q2+q3)/2.0))+(2.0*r*r))
          e71=(r*r)/((1-((q1+q2)/2.0))+(2.0*r*r))
*           e72=(r*r)/((1-((q2+q3)/2.0))+(2.0*r*r))
          e81=(r*r*h*h*(1.0+((q1+q2)/2.0)))/(2.0*((1.0-((q1+q2)/2))
     ++(2.0*r*r)))
*          e82=(r*r*h*h*(1.0+((q2+q3)/2.0)))/(2.0*((1.0-((q2+q3)/2))
*    ++(2.0*r*r)))

      i = intfc1
       do 592 j = 2, kkj-1

       IF(j .le. jjd)then
          x1=1.0
          x2=1.0
          x3=1.0
          i0=1
          j0=1
       else IF(j .gt. jjd)then
          x1=-1.0
```

```
            x2=-1.0
            x3=1.0
            i0=1
            j0=-1
         else
         endif

            m1 = a1(i,j)
            m2 = a1(i-i0,j)
            m3 = a1(i,j-j0)
            m4 = a1(i,j+j0)
            m5 = a1(i+i0,j)
            m6 = a1(i+i0,j+j0)
            m7 = a1(i,j+2*j0)
            m8 = a1(i+2*i0,j)
            m9 = a1(i+2*i0,j+j0)
            m10=a1(i+i0,j+2*j0)
            m11=a1(i+2*i0,j+2*j0)
            m12=a1(i-i0,j+j0)
            m13=a1(i-i0,j+2*j0)
            m14=a1(i,j+3*j0)
            m15=a1(i+i0,j-j0)
            m16=a1(i+2*i0,j-j0)
            m17=a1(i+3*i0,j)
*...........................................................
      uu(m1)=(9.0*si(m1)-12.0*si(m4)+3.0*si(m7)-12.0*si(m5)+16.0*si(m6)
     +-4.0*si(m10)+3.0*si(m8)-4.0*si(m9)+si(m11))/(4.0*r*h*h*x1)

      vv(m1)=(e71*(si(m2)+si(m5))+e61*(si(m3)+si(m4))-si(m1))/(-e81)

      ssgx(m1)=(si(m1)*(6.*zk9-5.)+si(m4)*(6.-8.*zk9)+
     +si(m7)*(2.*zk9-3.)+si(m14)*0.5+si(m3)*1.5+si(m2)*(-3.*zk9)+
     +si(m12)*4.*zk9+si(m13)*(-zk9)+si(m5)*(-3.*zk9)+si(m6)*4.*zk9+
     +si(m10)*(-zk9))*e/(p2*x2)

      ssgy(m1)=(si(m3)*(-1.5)+si(m1)*(5.0+6.0*e10)-si(m4)*(6.+8.*e10)
     ++si(m7)*(3.0+2.*e10)-si(m14)*0.5-si(m2)*3.*e10+si(m12)*4.*e10
     +-si(m13)*e10-si(m5)*3.*e10+si(m6)*4.*e10
     +-si(m10)*e10)*e/(-c*p2*x2)

      ssgxy(m1)=(si(m2)*3.*e8+si(m1)*(3.-10.*e8)-si(m3)*1.5-
     +si(m4)*1.5+si(m5)*(12.*e8-4.)+si(m15)*2.+si(m6)*2.+
     +si(m8)*(1.-6.*e8)-si(m16)*0.5-
     +si(m9)*0.5+si(m17)*e8)*(r*e/(p2*x3))
 592   continue
*  upper interface calculation finished

****  middle field  calculation started
         iib=(intfc2+intfc1)/2
         jjb=kkj/2
          e=er1
          c=q2
          c1=1.0-c
          c2=1.0+c
              zk6=(c1/2.0)/(c1+(2.0*r*r))
              zk7=(r*r)/(c1+(2.0*r*r))
              zk9=(r*r)/(2.0*c)
              zk10=(r*r*(2.0+c))/2.0
```

```
            zk8=(r*r*h*h*c2)/(2.0*(c1+(2.0*r*r)))
            p2=(c2*c2*r*r*r*h*h*h)/c

      do 93 i = intfc1+1,intfc2-1
       do 94 j = 2, kkj-1

       IF(i .LE. iib .and. j .le. jjb)then
          x1=1.0
          x2=1.0
          x3=1.0
          i0=1
          j0=1
       else IF(i .gt. iib .and. j .le. jjb)then
          x1=-1.0
          x2=1.0
          x3=-1.0
          i0=-1
          j0=1
       else IF(i .gt. iib .and. j .gt. jjb)then
          x1=1.0
          x2=-1.0
          x3=-1.0
          i0=-1
          j0=-1
       else IF(i .LE. iib .and. j .gt. jjb)then
          x1=-1.0
          x2=-1.0
          x3=1.0
          i0=1
          j0=-1
       else
       endif

          m1 = a1(i,j)
          m2 = a1(i-i0,j)
          m3 = a1(i,j-j0)
          m4 = a1(i,j+j0)
          m5 = a1(i+i0,j)
          m6 = a1(i+i0,j+j0)
          m7 = a1(i,j+2*j0)
          m8 = a1(i+2*i0,j)
          m9 = a1(i+2*i0,j+j0)
          m10=a1(i+i0,j+2*j0)
          m11=a1(i+2*i0,j+2*j0)
          m12=a1(i-i0,j+j0)
          m13=a1(i-i0,j+2*j0)
          m14=a1(i,j+3*j0)
          m15=a1(i+i0,j-j0)
          m16=a1(i+2*i0,j-j0)
          m17=a1(i+3*i0,j)

*............................................................
      uu(m1)=(9.0*si(m1)-12.0*si(m4)+3.0*si(m7)-12.0*si(m5)+16.0*si(m6)
     + -4.0*si(m10)+3.0*si(m8)-4.0*si(m9)+si(m11))/(4.0*r*h*h*x1)

      vv(m1)=(zk7*(si(m2)+si(m5))+zk6*(si(m3)+si(m4))-si(m1))/(-zk8)
```

```
     ssgx(m1)=(si(m1)*(6*zk9-5)+si(m4)*(6.-8.*zk9)+si(m7)*(2*zk9-3)
    ++si(m14)*0.5+si(m3)*1.5+si(m2)*(-3*zk9)+si(m12)*4.*zk9
    ++si(m13)*(-zk9)+si(m5)*(-3.*zk9)+si(m6)*4.*zk9
    ++si(m10)*(-zk9))*e*x2/(p2)

     ssgy(m1)=(si(m3)*(-1.5)+si(m1)*(5.0+6.0*zk10)-si(m4)*(6.+8.*zk10)
    +  +si(m7)*(3.0+2.*zk10)-si(m14)*0.5-si(m2)*3.*zk10+si(m12)*4.*zk10
    +  -si(m13)*zk10-si(m5)*3.*zk10+si(m6)*4.*zk10
    +  -si(m10)*zk10)*e/(-c*p2*x2)

     ssgxy(m1)=(si(m2)*3.*zk9+si(m1)*(3.-10.*zk9)-si(m3)*1.5-si(m4)*1.5
    +        +si(m5)*(12.*zk9-4.)+si(m15)*2.+si(m6)*2.+si(m8)*(1.-6.*zk9)
    +        -si(m16)*0.5-si(m9)*0.5+si(m17)*zk9)*(r*e/(p2*x3))
  94   continue
  93  continue
****  middle field calculation finished

****  lower interface  calculation started
*         iie=intfc2
         jje=kkj/2
          e=(er1+er2)/2
          c=(q2+q3)/2
          c1=1.0-c
          c2=1.0+c
              zk6=(c1/2.0)/(c1+(2.0*r*r))
              zk7=(r*r)/(c1+(2.0*r*r))
              zk9=(r*r)/(2.0*c)
              zk10=(r*r*(2.0+c))/2.0

              zk8=(r*r*h*h*c2)/(2.0*(c1+(2.0*r*r)))
              p2=(c2*c2*r*r*r*h*h*h)/c
*             e8=(r*r)/(2.0*((q1+q2)/2.0))
         e9=(r*r)/(2.0*((q2+q3)/2.0))
*         e10=(r*r*(2.0+((q1+q2)/2)))/2.0
         e11=(r*r*(2.0+((q2+q3)/2)))/2.0
*           e61=((1-((q1+q2)/2.0))/2.0)/((1-((q1+q2)/2.0))+(2.0*r*r))
           e62=((1-((q2+q3)/2.0))/2.0)/((1-((q2+q3)/2.0))+(2.0*r*r))
*           e71=(r*r)/((1-((q1+q2)/2.0))+(2.0*r*r))
           e72=(r*r)/((1-((q2+q3)/2.0))+(2.0*r*r))
*           e81=(r*r*h*h*(1.0+((q1+q2)/2.0)))/(2.0*((1.0-((q1+q2)/2))
*    ++(2.0*r*r)))
          e82=(r*r*h*h*(1.0+((q2+q3)/2.0)))/(2.0*((1.0-((q2+q3)/2))
    ++(2.0*r*r)))


      i = intfc2
       do 594 j = 2, kkj-1

       IF(j .le. jje)then
          x1=1.0
          x2=1.0
          x3=1.0
          i0=1
          j0=1
       else IF(j .gt. jje)then
          x1=-1.0
          x2=-1.0
          x3=+1.0
```

```
            i0=+1
            j0=-1
         else
         endif

            m1 = a1(i,j)
            m2 = a1(i-i0,j)
            m3 = a1(i,j-j0)
            m4 = a1(i,j+j0)
            m5 = a1(i+i0,j)
            m6 = a1(i+i0,j+j0)
            m7 = a1(i,j+2*j0)
            m8 = a1(i+2*i0,j)
            m9 = a1(i+2*i0,j+j0)
            m10=a1(i+i0,j+2*j0)
            m11=a1(i+2*i0,j+2*j0)
            m12=a1(i-i0,j+j0)
            m13=a1(i-i0,j+2*j0)
            m14=a1(i,j+3*j0)
            m15=a1(i+i0,j-j0)
            m16=a1(i+2*i0,j-j0)
            m17=a1(i+3*i0,j)

*...........................................................
      uu(m1)=(9.0*si(m1)-12.0*si(m4)+3.0*si(m7)-12.0*si(m5)+16.0*si(m6)
     + -4.0*si(m10)+3.0*si(m8)-4.0*si(m9)+si(m11))/(4.0*r*h*h*x1)

      vv(m1)=(e72*(si(m2)+si(m5))+e62*(si(m3)+si(m4))-si(m1))/(-e82)

      ssgx(m1)=(si(m1)*(6*zk9-5)+si(m4)*(6.-8.*zk9)+si(m7)*(2*zk9-3)
     ++si(m14)*0.5+si(m3)*1.5+si(m2)*(-3*zk9)+si(m12)*4.*zk9
     ++si(m13)*(-zk9)+si(m5)*(-3.*zk9)+si(m6)*4.*zk9
     ++si(m10)*(-zk9))*e*x2/(p2)

      ssgy(m1)=(si(m3)*(-1.5)+si(m1)*(5.0+6.0*e11)-si(m4)*(6.+8.*e11)
     +  +si(m7)*(3.0+2.*e11)-si(m14)*0.5-si(m2)*3.*e11+si(m12)*4.*e11
     +  -si(m13)*e11-si(m5)*3.*e11+si(m6)*4.*e11
     +  -si(m10)*e11)*e/(-c*p2*x2)

      ssgxy(m1)=(si(m2)*3.*e9+si(m1)*(3.-10.*e9)-si(m3)*1.5-si(m4)*1.5
     +        +si(m5)*(12.*e9-4.)+si(m15)*2.+si(m6)*2.+si(m8)*(1.-6.*e9)
     +        -si(m16)*0.5-si(m9)*0.5+si(m17)*e9)*(r*e/(p2*x3))
 594   continue
****  lower interface calculation finished

****  lower field  calculation started
         iic=(kki+intfc2)/2
         jjc=kkj/2
          e=er2
          c=q3
          c1=1.0-c
          c2=1.0+c
              zk6=(c1/2.0)/(c1+(2.0*r*r))
              zk7=(r*r)/(c1+(2.0*r*r))
              zk9=(r*r)/(2.0*c)
              zk10=(r*r*(2.0+c))/2.0

              zk8=(r*r*h*h*c2)/(2.0*(c1+(2.0*r*r)))
```

```
           p2=(c2*c2*r*r*r*h*h*h)/c

      do 95 i = intfc2+1,kki-1
       do 996 j = 2, kkj-1

       IF(i .LE. iic .and. j .le. jjc)then
          x1=1.0
          x2=1.0
          x3=1.0
          i0=1
          j0=1
       else IF(i .gt. iic .and. j .le. jjc)then
          x1=-1.0
          x2=1.0
          x3=-1.0
          i0=-1
          j0=1
       else IF(i .gt. iic .and. j .gt. jjc)then
          x1=1.0
          x2=-1.0
          x3=-1.0
          i0=-1
          j0=-1
       else IF(i .LE. iic .and. j .gt. jjc)then
          x1=-1.0
          x2=-1.0
          x3=1.0
          i0=1
          j0=-1
       else
       endif

          m1 = a1(i,j)
          m2 = a1(i-i0,j)
          m3 = a1(i,j-j0)
          m4 = a1(i,j+j0)
          m5 = a1(i+i0,j)
          m6 = a1(i+i0,j+j0)
          m7 = a1(i,j+2*j0)
          m8 = a1(i+2*i0,j)
          m9 = a1(i+2*i0,j+j0)
          m10=a1(i+i0,j+2*j0)
          m11=a1(i+2*i0,j+2*j0)
          m12=a1(i-i0,j+j0)
          m13=a1(i-i0,j+2*j0)
          m14=a1(i,j+3*j0)
          m15=a1(i+i0,j-j0)
          m16=a1(i+2*i0,j-j0)
          m17=a1(i+3*i0,j)

*..........................................................
      uu(m1)=(9.0*si(m1)-12.0*si(m4)+3.0*si(m7)-12.0*si(m5)+16.0*si(m6)
     + -4.0*si(m10)+3.0*si(m8)-4.0*si(m9)+si(m11))/(4.0*r*h*h*x1)

      vv(m1)=(zk7*(si(m2)+si(m5))+zk6*(si(m3)+si(m4))-si(m1))/(-zk8)

      ssgx(m1)=(si(m1)*(6*zk9-5)+si(m4)*(6.-8.*zk9)+si(m7)*(2*zk9-3)
     ++si(m14)*0.5+si(m3)*1.5+si(m2)*(-3*zk9)+si(m12)*4.*zk9
```

```
    ++si(m13)*(-zk9)+si(m5)*(-3.*zk9)+si(m6)*4.*zk9
    ++si(m10)*(-zk9))*e*x2/(p2)

     ssgy(m1)=(si(m3)*(-1.5)+si(m1)*(5.0+6.0*zk10)-si(m4)*(6.+8.*zk10)
    +  +si(m7)*(3.0+2.*zk10)-si(m14)*0.5-si(m2)*3.*zk10+si(m12)*4.*zk10
    +  -si(m13)*zk10-si(m5)*3.*zk10+si(m6)*4.*zk10
    +  -si(m10)*zk10)*e/(-c*p2*x2)

     ssgxy(m1)=(si(m2)*3.*zk9+si(m1)*(3.-10.*zk9)-si(m3)*1.5-si(m4)*1.5
    +        +si(m5)*(12.*zk9-4.)+si(m15)*2.+si(m6)*2.+si(m8)*(1.-6.*zk9)
    +        -si(m16)*0.5-si(m9)*0.5+si(m17)*zk9)*(r*e/(p2*x3))
  996  continue
  95   continue

  110 WRITE(*,*)'lyr = ',lyr

*      upper interface double calculation started     upper interface double calculation started
upper interface double calculation started
         c121=(q1-1)/((1+q1)*r*r*h*h)
         c131=-2/((1+q1)*h*h)
         c161=(2+q1)/((1+q1)*(1+q1)*2*r*h*h*h)
         c171=1/((1+q1)*(1+q1)*2*r*r*r*h*h*h)
         c181=q1/((1+q1)*(1+q1)*2*r*r*h*h*h)
         c191=1/((1+q1)*(1+q1)*2*h*h*h)
         d131=-6*c181-c191
         d141=8*c181+3*c191
         d151=-2*c181-3*c191

         c122=(q2-1)/((1+q2)*r*r*h*h)
         c132=-2/((1+q2)*h*h)
         c162=(2+q2)/((1+q2)*(1+q2)*2*r*h*h*h)
         c172=1/((1+q2)*(1+q2)*2*r*r*r*h*h*h)
         c182=q2/((1+q2)*(1+q2)*2*r*r*h*h*h)
         c192=1/((1+q2)*(1+q2)*2*h*h*h)
         d132=-6*c182-c192
         d142=8*c182+3*c192
         d152=-2*c182-3*c192

         c123=(q3-1)/((1+q3)*r*r*h*h)
         c133=-2/((1+q3)*h*h)
         c163=(2+q3)/((1+q3)*(1+q3)*2*r*h*h*h)
         c173=1/((1+q3)*(1+q3)*2*r*r*r*h*h*h)
         c183=q3/((1+q3)*(1+q3)*2*r*r*h*h*h)
         c193=1/((1+q3)*(1+q3)*2*h*h*h)
         d133=-6*c183-c193
         d143=8*c183+3*c193
         d153=-2*c183-3*c193

*..............needed only for Calc....for sigx calc...
         c141=1/((1+q1)*(1+q1)*2*r*h*h*h)
         c151=q1/((1+q1)*(1+q1)*2*r*r*r*h*h*h)

         c142=1/((1+q2)*(1+q2)*2*r*h*h*h)
         c152=q2/((1+q2)*(1+q2)*2*r*r*r*h*h*h)

         c143=1/((1+q3)*(1+q3)*2*r*h*h*h)
         c153=q2/((1+q3)*(1+q3)*2*r*r*r*h*h*h)
```

```
*.............needed only for Calc....for sigx calc...
         jjm=kkj/2

       i = intfc1
        do 591 j = 2, kkj-1

        IF(j .le. jjm)then
           x1=1.0
           i0=1
           j0=1
        else IF(j .gt. jjm)then
           x1=-1.0
           i0=1
           j0=-1
        else
        endif

            m1 = a1(i,j)
            m2 = a1(i,j+j0)
            m3 = a1(i,j-j0)
            m4 = a1(i-i0,j-j0)
            m5 = a1(i-i0,j)
            m6 = a1(i-i0,j+j0)
            m7 = a1(i-2*i0,j-j0)
            m8 = a1(i-2*i0,j)
            m9 = a1(i-2*i0,j+j0)
            m10 = a1(i-3*i0,j)
            m11 = a1(i+i0,j-j0)
            m12 = a1(i+i0,j)
            m13 = a1(i+i0,j+j0)
            m14 = a1(i+2*i0,j-j0)
            m15 = a1(i+2*i0,j)
            m16 = a1(i+2*i0,j+j0)
            m17 = a1(i+3*i0,j)
*             m18 = a1(i-3*i0,j-j0)
            m19 = a1(i-3*i0,j+j0)
*             m20 = a1(i+3*i0,j-j0)
            m21 = a1(i+3*i0,j+j0)
            m22 = a1(i,j+2*j0)
            m23 = a1(i+i0,j+2*j0)
            m24 = a1(i+2*i0,j+2*j0)
            m25 = a1(i-i0,j+2*j0)
            m26 = a1(i,j+3*j0)
            m27 = a1(i-2*i0,j+2*j0)
            m28 = a1(i-3*i0,j+2*j0)
            m29 = a1(i+3*i0,j+2*j0)


*  New Calc formula   for interface  Upperbody.....

      uiu(m1)=(1.0/(4.0*r*h*h))*x1*(-9.*si(m1)+12.*si(m2)-3.*si(m22)+
     +12.*si(m5)-16.*si(m6)+4.*si(m25)-3.*si(m8)+4.*si(m9)-si(m27))

      viu(m1)=(-2.*c121+2.*c131)*si(m1)+c121*si(m2)+c121*si(m3)-
     +  5.*c131*si(m5)+4.*c131*si(m8)-c131*si(m10)

      sxiu(m1)=x1*((-6.*c141-10.*c151)*si(m1)+(8.*c141+12.*c151)*si(m2)+
     +  3.*c151*si(m3)+(-2.*c141-6.*c151)*si(m22)+c151*si(m26)+
```

```
     +  15.*c141*si(m5)-20.*c141*si(m6)+5.*c141*si(m25)-12.*c141*si(m8)+
     +  16.*c141*si(m9)-4.*c141*si(m27)+3.*c141*si(m10)-4.*c141*si(m19)+
     +  c141*si(m28))

      syiu(m1)=x1*((6.*c161-10.*c171)*si(m1)+(-8.*c161+12.*c171)*si(m2)+
     +  3.*c171*si(m3)+(2.*c161-6.*c171)*si(m22)+c171*si(m26)-
     +  15.*c161*si(m5)+20.*c161*si(m6)-5.*c161*si(m25)+12.*c161*si(m8)-
     +  16.*c161*si(m9)+4.*c161*si(m27)-3.*c161*si(m10)+4.*c161*si(m19)-
     +  c161*si(m28))

      sxyiu(m1)=d131*si(m1)+3.*c181*si(m2)+3.*c181*si(m3)-
     +  4.*c181*si(m4)+d141*si(m5)-4.*c181*si(m6)+c181*si(m7)+
     +  d151*si(m8)+c181*si(m9)+c191*si(m10)

*  New Calc formula for interface Lowerbody.....

      uil(m1)=(1.0/(4.0*r*h*h))*x1*(9.*si(m1)-12.*si(m2)+3.*si(m22)-
     +12.*si(m12)+16.*si(m13)-4.*si(m23)+3.*si(m15)-4.*si(m16)+si(m24))

      vil(m1)=(-2.*c122+2.*c132)*si(m1)+c122*si(m2)+c122*si(m3)-
     +  5.*c132*si(m12)+4.*c132*si(m15)-c132*si(m17)

      sxil(m1)=x1*((-6.*c142-10.*c152)*si(m1)+(8.*c142+12.*c152)*si(m2)+
     +  3.*c152*si(m3)+(-2.*c142-6.*c152)*si(m22)+c152*si(m26)+
     +  15.*c142*si(m12)-20.*c142*si(m13)+5.*c142*si(m23)-
     +  12.*c142*si(m15)+16.*c142*si(m16)-
     +  4.*c142*si(m24)+3.*c142*si(m17)-4.*c142*si(m21)+
     +  c142*si(m29))*er1

      syil(m1)=x1*((6.*c162-10.*c172)*si(m1)+(-8.*c162+12.*c172)*si(m2)+
     +  3.*c172*si(m3)+(2.*c162-6.*c172)*si(m22)+c172*si(m26)-
     +  15.*c162*si(m12)+20.*c162*si(m13)-5.*c162*si(m23)+
     +  12.*c162*si(m15)-16.*c162*si(m16)+
     +  4.*c162*si(m24)-3.*c162*si(m17)+4.*c162*si(m21)-
     +  c162*si(m29))*er1

      sxyil(m1)=(-d132*si(m1)-3.*c182*si(m2)-3.*c182*si(m3)+
     +  4.*c182*si(m11)-d142*si(m12)+4.*c182*si(m13)-c182*si(m14)-
     +  d152*si(m15)-c182*si(m16)-c192*si(m17))*er1
      IF(m1 .EQ. 192) WRITE(*,*) m1, uiu(m1),uil(m1)
 591  continue
*  upper interface calculation finished

*      lower interface double calculation started        *      lower interface double calculation started
*      lower interface double calculation started

          jjn=kkj/2

        i = intfc2
         do 593 j = 2, kkj-1

         IF(j .le. jjn)then
            x1=1.0
            i0=1
            j0=1
         else IF(j .gt. jjn)then
            x1=-1.0
            i0=1
```

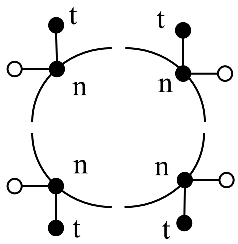
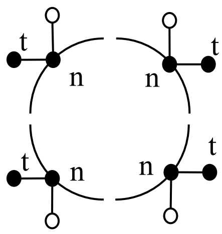
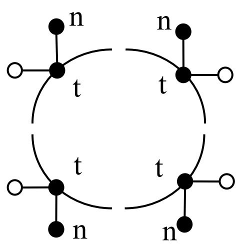
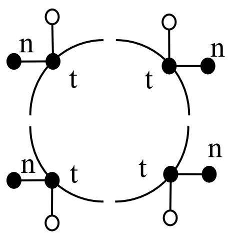
```
          j0=-1
        else
        endif


          m1 = a1(i,j)
          m2 = a1(i,j+j0)
          m3 = a1(i,j-j0)
          m4 = a1(i-i0,j-j0)
          m5 = a1(i-i0,j)
          m6 = a1(i-i0,j+j0)
          m7 = a1(i-2*i0,j-j0)
          m8 = a1(i-2*i0,j)
          m9 = a1(i-2*i0,j+j0)
          m10 = a1(i-3*i0,j)
          m11 = a1(i+i0,j-j0)
          m12 = a1(i+i0,j)
          m13 = a1(i+i0,j+j0)
          m14 = a1(i+2*i0,j-j0)
          m15 = a1(i+2*i0,j)
          m16 = a1(i+2*i0,j+j0)
          m17 = a1(i+3*i0,j)
*           m18 = a1(i-3*i0,j-j0)
          m19 = a1(i-3*i0,j+j0)
*           m20 = a1(i+3*i0,j-j0)
          m21 = a1(i+3*i0,j+j0)
          m22 = a1(i,j+2*j0)
          m23 = a1(i+i0,j+2*j0)
          m24 = a1(i+2*i0,j+2*j0)
          m25 = a1(i-i0,j+2*j0)
          m26 = a1(i,j+3*j0)
          m27 = a1(i-2*i0,j+2*j0)
          m28 = a1(i-3*i0,j+2*j0)
          m29 = a1(i+3*i0,j+2*j0)

*  New Calc formula for interface  Upperbody.....

      uit(m1)=(1.0/(4.0*r*h*h))*x1*(-9.*si(m1)+12.*si(m2)-3.*si(m22)+
     +12.*si(m5)-16.*si(m6)+4.*si(m25)-3.*si(m8)+4.*si(m9)-si(m27))

      vit(m1)=(-2.*c122+2.*c132)*si(m1)+c122*si(m2)+c122*si(m3)-
     +  5.*c132*si(m5)+4.*c132*si(m8)-c132*si(m10)

      sxit(m1)=x1*((-6.*c142-10.*c152)*si(m1)+(8.*c142+12.*c152)*si(m2)+
     +  3.*c152*si(m3)+(-2.*c142-6.*c152)*si(m22)+c152*si(m26)+
     +  15.*c142*si(m5)-20.*c142*si(m6)+5.*c142*si(m25)-12.*c142*si(m8)+
     +  16.*c142*si(m9)-4.*c142*si(m27)+3.*c142*si(m10)-4.*c142*si(m19)+
     +  c142*si(m28))*er1

      syit(m1)=x1*((6.*c162-10.*c172)*si(m1)+(-8.*c162+12.*c172)*si(m2)+
     +  3.*c172*si(m3)+(2.*c162-6.*c172)*si(m22)+c172*si(m26)-
     +  15.*c162*si(m5)+20.*c162*si(m6)-5.*c162*si(m25)+12.*c162*si(m8)-
     +  16.*c162*si(m9)+4.*c162*si(m27)-3.*c162*si(m10)+4.*c162*si(m19)-
     +  c162*si(m28))*er1

      sxyit(m1)=(d132*si(m1)+3.*c182*si(m2)+3.*c182*si(m3)-
     +  4.*c182*si(m4)+d142*si(m5)-4.*c182*si(m6)+c182*si(m7)+
     +  d152*si(m8)+c182*si(m9)+c192*si(m10))*er1
```

```
*  New Calc formula for interface Lowerbody.....

      uib(m1)=(1.0/(4.0*r*h*h))*x1*(9.*si(m1)-12.*si(m2)+3.*si(m22)-
     +12.*si(m12)+16.*si(m13)-4.*si(m23)+3.*si(m15)-4.*si(m16)+si(m24))

      vib(m1)=(-2.*c123+2.*c133)*si(m1)+c123*si(m2)+c123*si(m3)-
     +  5.*c133*si(m12)+4.*c133*si(m15)-c133*si(m17)

      sxib(m1)=x1*((-6.*c143-10.*c153)*si(m1)+(8.*c143+12.*c153)*si(m2)+
     +  3.*c153*si(m3)+(-2.*c143-6.*c153)*si(m22)+c153*si(m26)+
     +  15.*c143*si(m12)-20.*c143*si(m13)+5.*c143*si(m23)-
     +  12.*c143*si(m15)+16.*c143*si(m16)-
     +  4.*c143*si(m24)+3.*c143*si(m17)-4.*c143*si(m21)+
     +  c143*si(m29))*er2

      syib(m1)=x1*((6.*c163-10.*c173)*si(m1)+(-8.*c163+12.*c173)*si(m2)+
     +  3.*c173*si(m3)+(2.*c163-6.*c173)*si(m22)+c173*si(m26)-
     +  15.*c163*si(m12)+20.*c163*si(m13)-5.*c163*si(m23)+
     +  12.*c163*si(m15)-16.*c163*si(m16)+
     +  4.*c163*si(m24)-3.*c163*si(m17)+4.*c163*si(m21)-
     +  c163*si(m29))*er2

      sxyib(m1)=(-d133*si(m1)-3.*c183*si(m2)-3.*c183*si(m3)+
     +  4.*c183*si(m11)-d143*si(m12)+4.*c183*si(m13)-c183*si(m14)-
     +  d153*si(m15)-c183*si(m16)-c193*si(m17))*er2

      IF(m1.EQ.1627)WRITE(*,616)m1,c142,c143,c152,c153,sxit(m1),sxib(m1)
      IF(m1.EQ.1628)WRITE(*,616)m1,c142,c143,c152,c153,sxit(m1),sxib(m1)
      IF(m1.EQ.1629)WRITE(*,616)m1,c142,c143,c152,c153,sxit(m1),sxib(m1)
 616  FORMAT(1x, i5,6e12.4)
 593  continue
*  lower interface calculation finished

      return
      end
```

APPENDIX-A

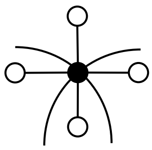
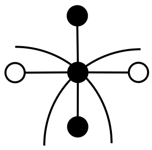
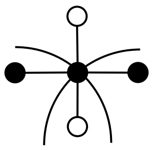
Tables required for input data files

Table A1: Selection of boundary conditions

ktb		kpc		kdc		ktc	
value	control	value	control	value	control	value	control
1	u_n, u_t	1		0		-1	n
2	u_n, σ_t	2		1		1	t
3	σ_n, u_t	3		-1		0	n, t
4	σ_n, σ_t	4				2	(skip)

APPENDIX-A

Table A2: Selection of boundary conditions for excess points

<table>
<tr><td rowspan="2">it</td><td colspan="2">ipc</td><td colspan="2">idc</td><td colspan="2">ivs</td></tr>
<tr><td>value</td><td>control</td><td>value</td><td>control</td><td>value</td><td>control</td></tr>
<tr><td rowspan="6">Serial number of the boundary point corresponding to the additional or previously omitted boundary or for given ψ values</td><td rowspan="2">1</td><td rowspan="2"></td><td rowspan="2">0</td><td rowspan="2">lp=l(ij+0)
mp=m(ij+0)</td><td>1</td><td>u_t</td></tr>
<tr><td>2</td><td>u_n</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2"></td><td rowspan="2">1</td><td rowspan="2">lp=l(ij+1)
mp=m(ij+1)</td><td>3</td><td>σ_n</td></tr>
<tr><td>4</td><td>σ_t</td></tr>
<tr><td rowspan="2">3</td><td rowspan="2"></td><td rowspan="2">-1</td><td rowspan="2">lp=l(ij-1)
mp=m(ij-1)</td><td rowspan="2">5</td><td rowspan="2">Declaration of ψ values</td></tr>
<tr></tr>
</table>

[Note: For comprehensive understanding of these two tables (Table A1 and A2) reference [27] can be looked into]

M. Sc.
Engg. (ME)
Thesis

MD. KHAIRUL HABIB PULOK

BUET
July
2016